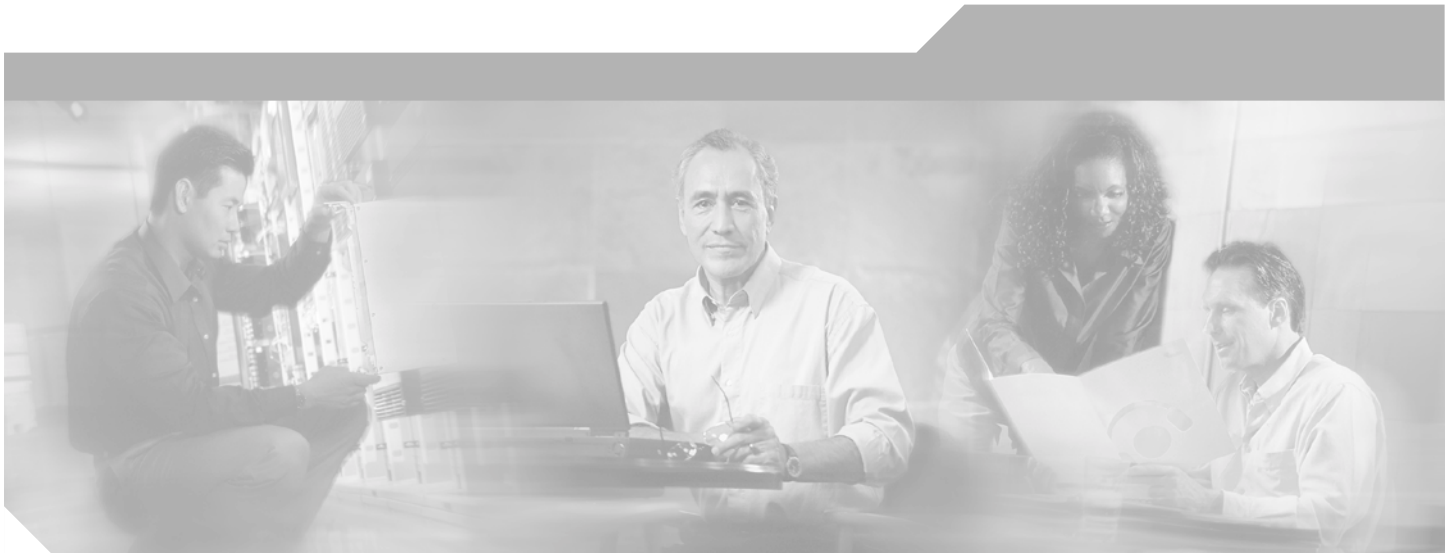


Cisco ME 3400 Ethernet Access Switch Software Configuration Guide

Cisco IOS Release 12.2(25)EX
November 2005

Corporate Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 526-4100

Customer Order Number: DOC-7817058=
Text Part Number: 78-17058-01

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

CCSP, CCVP, the Cisco Square Bridge logo, Follow Me Browsing, and StackWise are trademarks of Cisco Systems, Inc.; Changing the Way We Work, Live, Play, and Learn, and iQuick Study are service marks of Cisco Systems, Inc.; and Access Registrar, Aironet, ASIST, BPX, Catalyst, CCDA, CCDP, CCIE, CCIP, CCNA, CCNP, Cisco, the Cisco Certified Internetwork Expert logo, Cisco IOS, Cisco Press, Cisco Systems, Cisco Systems Capital, the Cisco Systems logo, Cisco Unity, Empowering the Internet Generation, Enterprise/Solver, EtherChannel, EtherFast, EtherSwitch, Fast Step, FormShare, GigaDrive, GigaStack, HomeLink, Internet Quotient, IOS, IP/TV, iQ Expertise, the iQ logo, iQ Net Readiness Scorecard, LightStream, Linksys, MeetingPlace, MGX, the Networkers logo, Networking Academy, Network Registrar, *Packet*, PIX, Post-Routing, Pre-Routing, ProConnect, RateMUX, ScriptShare, SlideCast, SMARTnet, StrataView Plus, TeleRouter, The Fastest Way to Increase Your Internet Quotient, and TransPath are registered trademarks of Cisco Systems, Inc. and/or its affiliates in the United States and certain other countries.

All other trademarks mentioned in this document or Website are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (0502R)

Cisco ME 3400 Ethernet Access Switch Software Configuration Guide
© 2005 Cisco Systems, Inc. All rights reserved.



CONTENTS

Preface xxxiii

- Audience xxxiii
- Purpose xxxiii
- Conventions xxxiii
- Related Publications xxxiv
- Obtaining Documentation xxxv
 - Cisco.com xxxv
 - Product Documentation DVD xxxv
 - Ordering Documentation xxxv
- Documentation Feedback xxxvi
- Cisco Product Security Overview xxxvi
 - Reporting Security Problems in Cisco Products xxxvii
- Obtaining Technical Assistance xxxvii
 - Cisco Technical Support & Documentation Website xxxvii
 - Submitting a Service Request xxxviii
 - Definitions of Service Request Severity xxxviii
- Obtaining Additional Publications and Information xxxix

CHAPTER 1

Overview 1-1

- Features 1-1
 - Performance Features 1-2
 - Management Options 1-3
 - Manageability Features 1-3
 - Availability Features 1-4
 - VLAN Features 1-5
 - Security Features 1-5
 - Subscriber Security 1-5
 - Switch Security 1-5
 - Network Security 1-6
 - Quality of Service and Class of Service Features 1-6
 - Layer 2 Virtual Private Network Services 1-7
 - Layer 3 Features 1-7
 - Layer 3 VPN Services 1-8
 - Monitoring Features 1-8

Default Settings After Initial Switch Configuration 1-8
Network Configuration Examples 1-11
Multidwelling or Ethernet-to-the-Subscriber Network 1-11
Layer 2 VPN Application 1-13
Multi-VRF CE Application 1-14
Where to Go Next 1-15

CHAPTER 2 **Using the Command-Line Interface 2-1**
Understanding Command Modes 2-1
Understanding the Help System 2-3
Understanding Abbreviated Commands 2-3
Understanding no and default Forms of Commands 2-4
Understanding CLI Error Messages 2-4
Using Command History 2-4
Changing the Command History Buffer Size 2-5
Recalling Commands 2-5
Disabling the Command History Feature 2-5
Using Editing Features 2-6
Enabling and Disabling Editing Features 2-6
Editing Commands through Keystrokes 2-6
Editing Command Lines that Wrap 2-8
Searching and Filtering Output of show and more Commands 2-8
Accessing the CLI 2-9
Accessing the CLI through a Console Connection or through Telnet 2-9

CHAPTER 3 **Assigning the Switch IP Address and Default Gateway 3-1**
Understanding the Boot Process 3-1
Assigning Switch Information 3-2
Default Switch Information 3-3
Understanding DHCP-Based Autoconfiguration 3-3
DHCP Client Request Process 3-3
Configuring DHCP-Based Autoconfiguration 3-5
DHCP Server Configuration Guidelines 3-5
Configuring the TFTP Server 3-5
Configuring the DNS 3-6
Configuring the Relay Device 3-6
Obtaining Configuration Files 3-7
Example Configuration 3-8

Manually Assigning IP Information 3-9
Checking and Saving the Running Configuration 3-10
Modifying the Startup Configuration 3-12
Default Boot Configuration 3-13
Automatically Downloading a Configuration File 3-13
Specifying the Filename to Read and Write the System Configuration 3-13
Booting Manually 3-14
Booting a Specific Software Image 3-14
Controlling Environment Variables 3-15
Scheduling a Reload of the Software Image 3-16
Configuring a Scheduled Reload 3-17
Displaying Scheduled Reload Information 3-18

CHAPTER 4 **Configuring Cisco IOS CNS Agents 4-1**
Understanding Cisco Configuration Engine Software 4-1
Configuration Service 4-2
Event Service 4-3
NameSpace Mapper 4-3
What You Should Know About the CNS IDs and Device Hostnames 4-3
ConfigID 4-3
DeviceID 4-4
Hostname and DeviceID 4-4
Using Hostname, DeviceID, and ConfigID 4-4
Understanding Cisco IOS Agents 4-5
Initial Configuration 4-5
Incremental (Partial) Configuration 4-6
Synchronized Configuration 4-6
Configuring Cisco IOS Agents 4-6
Enabling Automated CNS Configuration 4-6
Enabling the CNS Event Agent 4-8
Enabling the Cisco IOS CNS Agent 4-9
Enabling an Initial Configuration 4-9
Enabling a Partial Configuration 4-11
Displaying CNS Configuration 4-12

CHAPTER 5 **Administering the Switch 5-1**
Managing the System Time and Date 5-1
Understanding the System Clock 5-2
Understanding Network Time Protocol 5-2

Configuring NTP 5-4
Default NTP Configuration 5-4
Configuring NTP Authentication 5-5
Configuring NTP Associations 5-6
Configuring NTP Broadcast Service 5-7
Configuring NTP Access Restrictions 5-8
Configuring the Source IP Address for NTP Packets 5-10
Displaying the NTP Configuration 5-11
Configuring Time and Date Manually 5-11
Setting the System Clock 5-11
Displaying the Time and Date Configuration 5-12
Configuring the Time Zone 5-12
Configuring Summer Time (Daylight Saving Time) 5-13
Configuring a System Name and Prompt 5-14
Default System Name and Prompt Configuration 5-15
Configuring a System Name 5-15
Understanding DNS 5-15
Default DNS Configuration 5-16
Setting Up DNS 5-16
Displaying the DNS Configuration 5-17
Creating a Banner 5-17
Default Banner Configuration 5-17
Configuring a Message-of-the-Day Login Banner 5-18
Configuring a Login Banner 5-19
Managing the MAC Address Table 5-19
Building the Address Table 5-20
MAC Addresses and VLANs 5-20
Default MAC Address Table Configuration 5-21
Changing the Address Aging Time 5-21
Removing Dynamic Address Entries 5-22
Configuring MAC Address Notification Traps 5-22
Adding and Removing Static Address Entries 5-24
Configuring Unicast MAC Address Filtering 5-25
Disabling MAC Address Learning on a VLAN 5-26
Displaying Address Table Entries 5-28
Managing the ARP Table 5-28

CHAPTER 6 **Configuring SDM Templates** 6-1

Understanding the SDM Templates 6-1

Configuring the Switch SDM Template 6-2

Default SDM Template 6-2

SDM Template Configuration Guidelines 6-2

Setting the SDM Template 6-3

Displaying the SDM Templates 6-4

CHAPTER 7 **Configuring Switch-Based Authentication** 7-1

Preventing Unauthorized Access to Your Switch 7-1

Protecting Access to Privileged EXEC Commands 7-2

Default Password and Privilege Level Configuration 7-2

Setting or Changing a Static Enable Password 7-3

Protecting Enable and Enable Secret Passwords with Encryption 7-4

Disabling Password Recovery 7-5

Setting a Telnet Password for a Terminal Line 7-6

Configuring Username and Password Pairs 7-7

Configuring Multiple Privilege Levels 7-8

Setting the Privilege Level for a Command 7-8

Changing the Default Privilege Level for Lines 7-9

Logging into and Exiting a Privilege Level 7-10

Controlling Switch Access with TACACS+ 7-10

Understanding TACACS+ 7-10

TACACS+ Operation 7-12

Configuring TACACS+ 7-13

Default TACACS+ Configuration 7-13

Identifying the TACACS+ Server Host and Setting the Authentication Key 7-13

Configuring TACACS+ Login Authentication 7-14

Configuring TACACS+ Authorization for Privileged EXEC Access and Network Services 7-16

Starting TACACS+ Accounting 7-17

Displaying the TACACS+ Configuration 7-17

Controlling Switch Access with RADIUS 7-18

Understanding RADIUS 7-18

RADIUS Operation 7-19

Configuring RADIUS 7-20
Default RADIUS Configuration 7-20
Identifying the RADIUS Server Host 7-20
Configuring RADIUS Login Authentication 7-23
Defining AAA Server Groups 7-25
Configuring RADIUS Authorization for User Privileged Access and Network Services 7-27
Starting RADIUS Accounting 7-28
Configuring Settings for All RADIUS Servers 7-29
Configuring the Switch to Use Vendor-Specific RADIUS Attributes 7-29
Configuring the Switch for Vendor-Proprietary RADIUS Server Communication 7-31
Displaying the RADIUS Configuration 7-31
Controlling Switch Access with Kerberos 7-32
Understanding Kerberos 7-32
Kerberos Operation 7-34
Authenticating to a Boundary Switch 7-35
Obtaining a TGT from a KDC 7-35
Authenticating to Network Services 7-35
Configuring Kerberos 7-36
Configuring the Switch for Local Authentication and Authorization 7-36
Configuring the Switch for Secure Shell 7-37
Understanding SSH 7-38
SSH Servers, Integrated Clients, and Supported Versions 7-38
Limitations 7-38
Configuring SSH 7-39
Configuration Guidelines 7-39
Setting Up the Switch to Run SSH 7-39
Configuring the SSH Server 7-40
Displaying the SSH Configuration and Status 7-41

CHAPTER 8 **Configuring IEEE 802.1x Port-Based Authentication 8-1**
Understanding IEEE 802.1x Port-Based Authentication 8-1
Device Roles 8-2
Authentication Initiation and Message Exchange 8-3
Ports in Authorized and Unauthorized States 8-4
IEEE 802.1x Accounting 8-5
IEEE 802.1x Accounting Attribute-Value Pairs 8-5
IEEE 802.1x Host Mode 8-6
Using IEEE 802.1x with Port Security 8-7
Using IEEE 802.1x with VLAN Assignment 8-8

Configuring IEEE 802.1x Authentication 8-9
Default IEEE 802.1x Configuration 8-9
IEEE 802.1x Configuration Guidelines 8-10
Configuring IEEE 802.1x Authentication 8-11
Configuring the Switch-to-RADIUS-Server Communication 8-12
Configuring Periodic Re-Authentication 8-13
Manually Re-Authenticating a Client Connected to a Port 8-14
Changing the Quiet Period 8-14
Changing the Switch-to-Client Retransmission Time 8-15
Setting the Switch-to-Client Frame-Retransmission Number 8-15
Setting the Re-Authentication Number 8-16
Configuring the Host Mode 8-17
Resetting the IEEE 802.1x Configuration to the Default Values 8-17
Configuring IEEE 802.1x Accounting 8-18
Displaying IEEE 802.1x Statistics and Status 8-19

CHAPTER 9 **Configuring Interface Characteristics 9-1**
Understanding Interface Types 9-1
Port-Based VLANs 9-2
Switch Ports 9-2
UNI and NNI Ports 9-3
Access Ports 9-3
Trunk Ports 9-4
Tunnel Ports 9-4
Routed Ports 9-4
Switch Virtual Interfaces 9-5
EtherChannel Port Groups 9-6
Connecting Interfaces 9-6
Using Interface Configuration Mode 9-7
Procedures for Configuring Interfaces 9-7
Configuring a Range of Interfaces 9-8
Configuring and Using Interface Range Macros 9-10
Configuring Ethernet Interfaces 9-11
Default Ethernet Interface Configuration 9-12
Configuring User Network and Network Node Interfaces 9-13
Configuring Interface Speed and Duplex Mode 9-14
Speed and Duplex Configuration Guidelines 9-15
Setting the Interface Speed and Duplex Parameters 9-15
Configuring IEEE 802.3x Flow Control 9-17

Configuring Auto-MDIX on an Interface 9-18
Adding a Description for an Interface 9-19
Configuring Layer 3 Interfaces 9-20
Configuring the System MTU 9-21
Monitoring and Maintaining the Interfaces 9-22
Monitoring Interface Status 9-23
Clearing and Resetting Interfaces and Counters 9-24
Shutting Down and Restarting the Interface 9-24

CHAPTER 10 **Configuring Command Macros** 10-1
Understanding Command Macros 10-1
Configuring Command Macros 10-2
Default Command Macro Configuration 10-2
Command Macro Configuration Guidelines 10-2
Creating Command Macros 10-3
Applying Command Macros 10-4
Displaying Command Macros 10-5

CHAPTER 11 **Configuring VLANs** 11-1
Understanding VLANs 11-1
Supported VLANs 11-3
Normal-Range VLANs 11-3
Extended-Range VLANs 11-4
VLAN Port Membership Modes 11-4
UNI VLANs 11-5
Creating and Modifying VLANs 11-6
Default Ethernet VLAN Configuration 11-7
VLAN Configuration Guidelines 11-8
Creating or Modifying an Ethernet VLAN 11-9
Assigning Static-Access Ports to a VLAN 11-10
Creating an Extended-Range VLAN with an Internal VLAN ID 11-11
Configuring UNI VLANs 11-12
Configuration Guidelines 11-12
Configuring UNI VLANs 11-13
Displaying VLANs 11-14
Configuring VLAN Trunks 11-14
Trunking Overview 11-14
IEEE 802.1Q Configuration Considerations 11-15

Default Layer 2 Ethernet Interface VLAN Configuration 11-16
Configuring an Ethernet Interface as a Trunk Port 11-16
Interaction with Other Features 11-16
Configuring a Trunk Port 11-17
Defining the Allowed VLANs on a Trunk 11-17
Configuring the Native VLAN for Untagged Traffic 11-19
Configuring Trunk Ports for Load Sharing 11-19
Load Sharing Using STP Port Priorities 11-20
Load Sharing Using STP Path Cost 11-21
Configuring VMPS 11-23
Understanding VMPS 11-23
Dynamic-Access Port VLAN Membership 11-24
Default VMPS Client Configuration 11-24
VMPS Configuration Guidelines 11-25
Configuring the VMPS Client 11-25
Entering the IP Address of the VMPS 11-25
Configuring Dynamic-Access Ports on VMPS Clients 11-26
Reconfirming VLAN Memberships 11-26
Changing the Reconfirmation Interval 11-27
Changing the Retry Count 11-27
Monitoring the VMPS 11-27
Troubleshooting Dynamic-Access Port VLAN Membership 11-28
VMPS Configuration Example 11-28

CHAPTER 12 **Configuring Private VLANs 12-1**
Understanding Private VLANs 12-1
Types of Private VLANs and Private-VLAN Ports 12-2
IP Addressing Scheme with Private VLANs 12-4
Private VLANs across Multiple Switches 12-4
Private VLANs and Unicast, Broadcast, and Multicast Traffic 12-5
Private VLANs and SVIs 12-5
Configuring Private VLANs 12-5
Tasks for Configuring Private VLANs 12-6
Default Private-VLAN Configuration 12-6
Private-VLAN Configuration Guidelines 12-6
Secondary and Primary VLAN Configuration 12-7
Private-VLAN Port Configuration 12-8
Limitations with Other Features 12-9
Configuring and Associating VLANs in a Private VLAN 12-10

Configuring a Layer 2 Interface as a Private-VLAN Host Port 12-12
Configuring a Layer 2 Interface as a Private-VLAN Promiscuous Port 12-13
Mapping Secondary VLANs to a Primary VLAN Layer 3 VLAN Interface 12-14
Monitoring Private VLANs 12-15

CHAPTER 13 **Configuring IEEE 802.1Q and Layer 2 Protocol Tunneling 13-1**
Understanding IEEE 802.1Q Tunneling 13-1
Configuring IEEE 802.1Q Tunneling 13-4
Default IEEE 802.1Q Tunneling Configuration 13-4
IEEE 802.1Q Tunneling Configuration Guidelines 13-4
Native VLANs 13-4
System MTU 13-5
IEEE 802.1Q Tunneling and Other Features 13-6
Configuring an IEEE 802.1Q Tunneling Port 13-6
Understanding Layer 2 Protocol Tunneling 13-8
Configuring Layer 2 Protocol Tunneling 13-10
Default Layer 2 Protocol Tunneling Configuration 13-11
Layer 2 Protocol Tunneling Configuration Guidelines 13-11
Configuring Layer 2 Protocol Tunneling 13-12
Configuring Layer 2 Tunneling for EtherChannels 13-14
Configuring the SP Edge Switch 13-14
Configuring the Customer Switch 13-15
Monitoring and Maintaining Tunneling Status 13-18

CHAPTER 14 **Configuring STP 14-1**
Understanding Spanning-Tree Features 14-1
STP Overview 14-2
Spanning-Tree Topology and BPDUs 14-3
Bridge ID, Switch Priority, and Extended System ID 14-4
Spanning-Tree Interface States 14-4
Blocking State 14-6
Listening State 14-6
Learning State 14-6
Forwarding State 14-7
Disabled State 14-7
How a Switch or Port Becomes the Root Switch or Root Port 14-7
Spanning Tree and Redundant Connectivity 14-8
Spanning-Tree Address Management 14-8
Accelerated Aging to Retain Connectivity 14-9

Spanning-Tree Modes and Protocols 14-9
Supported Spanning-Tree Instances 14-10
Spanning-Tree Interoperability and Backward Compatibility 14-10
STP and IEEE 802.1Q Trunks 14-10
Configuring Spanning-Tree Features 14-11
Default Spanning-Tree Configuration 14-11
Spanning-Tree Configuration Guidelines 14-12
Changing the Spanning-Tree Mode. 14-13
Disabling Spanning Tree 14-14
Configuring the Root Switch 14-14
Configuring a Secondary Root Switch 14-16
Configuring Port Priority 14-16
Configuring Path Cost 14-18
Configuring the Switch Priority of a VLAN 14-19
Configuring Spanning-Tree Timers 14-20
Configuring the Hello Time 14-20
Configuring the Forwarding-Delay Time for a VLAN 14-21
Configuring the Maximum-Aging Time for a VLAN 14-21
Displaying the Spanning-Tree Status 14-22

CHAPTER 15 **Configuring MSTP** 15-1
Understanding MSTP 15-2
Multiple Spanning-Tree Regions 15-2
IST, CIST, and CST 15-2
Operations Within an MST Region 15-3
Operations Between MST Regions 15-3
Hop Count 15-4
Boundary Ports 15-5
Interoperability with IEEE 802.1D STP 15-5
Understanding RSTP 15-6
Port Roles and the Active Topology 15-6
Rapid Convergence 15-7
Synchronization of Port Roles 15-8
Bridge Protocol Data Unit Format and Processing 15-9
Processing Superior BPDU Information 15-10
Processing Inferior BPDU Information 15-10
Topology Changes 15-10

Configuring MSTP Features 15-11
Default MSTP Configuration 15-12
MSTP Configuration Guidelines 15-12
Specifying the MST Region Configuration and Enabling MSTP 15-13
Configuring the Root Switch 15-14
Configuring a Secondary Root Switch 15-16
Configuring Port Priority 15-17
Configuring Path Cost 15-18
Configuring the Switch Priority 15-19
Configuring the Hello Time 15-19
Configuring the Forwarding-Delay Time 15-20
Configuring the Maximum-Aging Time 15-21
Configuring the Maximum-Hop Count 15-21
Specifying the Link Type to Ensure Rapid Transitions 15-22
Restarting the Protocol Migration Process 15-22
Displaying the MST Configuration and Status 15-23

CHAPTER 16 **Configuring Optional Spanning-Tree Features 16-1**
Understanding Optional Spanning-Tree Features 16-1
Understanding Port Fast 16-2
Understanding BPDU Guard 16-2
Understanding BPDU Filtering 16-3
Understanding EtherChannel Guard 16-3
Understanding Root Guard 16-3
Understanding Loop Guard 16-4
Configuring Optional Spanning-Tree Features 16-5
Default Optional Spanning-Tree Configuration 16-5
Optional Spanning-Tree Configuration Guidelines 16-5
Enabling Port Fast 16-5
Enabling BPDU Guard 16-6
Enabling BPDU Filtering 16-7
Enabling EtherChannel Guard 16-8
Enabling Root Guard 16-9
Enabling Loop Guard 16-9
Displaying the Spanning-Tree Status 16-10

CHAPTER 17 **Configuring Flex Links** 17-1

Understanding Flex Links 17-1

Configuring Flex Links 17-2

Default Flex Link Configuration 17-2

Flex Link Configuration Guidelines 17-2

Configuring Flex Links 17-3

Monitoring Flex Links 17-4

CHAPTER 18 **Configuring DHCP Features and IP Source Guard** 18-1

Understanding DHCP Features 18-1

DHCP Server 18-2

DHCP Relay Agent 18-2

DHCP Snooping 18-2

Option-82 Data Insertion 18-3

DHCP Snooping Binding Database 18-5

Configuring DHCP Features 18-6

Default DHCP Configuration 18-7

DHCP Snooping Configuration Guidelines 18-7

Configuring the DHCP Relay Agent 18-8

Specifying the Packet Forwarding Address 18-9

Enabling DHCP Snooping and Option 82 18-10

Enabling DHCP Snooping on Private VLANs 18-11

Enabling the DHCP Snooping Binding Database Agent 18-12

Displaying DHCP Snooping Information 18-13

Understanding IP Source Guard 18-13

Source IP Address Filtering 18-14

Source IP and MAC Address Filtering 18-14

Configuring IP Source Guard 18-14

Default IP Source Guard Configuration 18-14

IP Source Guard Configuration Guidelines 18-15

Enabling IP Source Guard 18-15

Displaying IP Source Guard Information 18-16

CHAPTER 19 **Configuring Dynamic ARP Inspection** 19-1

Understanding Dynamic ARP Inspection 19-1
Interface Trust States and Network Security 19-3
Rate Limiting of ARP Packets 19-4
Relative Priority of ARP ACLs and DHCP Snooping Entries 19-4
Logging of Dropped Packets 19-4
Configuring Dynamic ARP Inspection 19-5
Default Dynamic ARP Inspection Configuration 19-5
Dynamic ARP Inspection Configuration Guidelines 19-6
Configuring Dynamic ARP Inspection in DHCP Environments 19-7
Configuring ARP ACLs for Non-DHCP Environments 19-8
Limiting the Rate of Incoming ARP Packets 19-10
Performing Validation Checks 19-12
Configuring the Log Buffer 19-13
Displaying Dynamic ARP Inspection Information 19-14

CHAPTER 20 **Configuring IGMP Snooping and MVR** 20-1

Understanding IGMP Snooping 20-1
IGMP Versions 20-2
Joining a Multicast Group 20-3
Leaving a Multicast Group 20-5
Immediate Leave 20-5
IGMP Configurable-Leave Timer 20-5
IGMP Report Suppression 20-6
Configuring IGMP Snooping 20-6
Default IGMP Snooping Configuration 20-6
Enabling or Disabling IGMP Snooping 20-7
Configuring a Multicast Router Port 20-8
Configuring a Host Statically to Join a Group 20-8
Enabling IGMP Immediate Leave 20-9
Configuring the IGMP Leave Timer 20-10
Configuring TCN-Related Commands 20-11
Controlling the Multicast Flooding Time After a TCN Event 20-11
Recovering from Flood Mode 20-11
Disabling Multicast Flooding During a TCN Event 20-12
Configuring the IGMP Snooping Querier 20-13
Disabling IGMP Report Suppression 20-14
Displaying IGMP Snooping Information 20-15

Understanding Multicast VLAN Registration 20-16
Using MVR in a Multicast Television Application 20-16
Configuring MVR 20-18
Default MVR Configuration 20-18
MVR Configuration Guidelines and Limitations 20-19
Configuring MVR Global Parameters 20-19
Configuring MVR Interfaces 20-20
Displaying MVR Information 20-22
Configuring IGMP Filtering and Throttling 20-22
Default IGMP Filtering and Throttling Configuration 20-23
Configuring IGMP Profiles 20-23
Applying IGMP Profiles 20-25
Setting the Maximum Number of IGMP Groups 20-25
Configuring the IGMP Throttling Action 20-26
Displaying IGMP Filtering and Throttling Configuration 20-28

CHAPTER 21 **Configuring Port-Based Traffic Control 21-1**
Configuring Storm Control 21-1
Understanding Storm Control 21-1
Default Storm Control Configuration 21-3
Configuring Storm Control and Threshold Levels 21-3
Configuring Protected Ports 21-5
Default Protected Port Configuration 21-5
Protected Port Configuration Guidelines 21-6
Configuring a Protected Port 21-6
Configuring Port Blocking 21-7
Default Port Blocking Configuration 21-7
Blocking Flooded Traffic on an Interface 21-7
Configuring Port Security 21-8
Understanding Port Security 21-8
Secure MAC Addresses 21-8
Security Violations 21-9
Default Port Security Configuration 21-10
Port Security Configuration Guidelines 21-10
Enabling and Configuring Port Security 21-11
Enabling and Configuring Port Security Aging 21-15
Displaying Port-Based Traffic Control Settings 21-17

CHAPTER 22 **Configuring CDP** 22-1

Understanding CDP 22-1

Configuring CDP 22-2

Default CDP Configuration 22-2

Configuring the CDP Characteristics 22-2

Disabling and Enabling CDP 22-3

Disabling and Enabling CDP on an Interface 22-4

Monitoring and Maintaining CDP 22-5

CHAPTER 23 **Configuring UDLD** 23-1

Understanding UDLD 23-1

Modes of Operation 23-1

Methods to Detect Unidirectional Links 23-2

Configuring UDLD 23-4

Default UDLD Configuration 23-4

Configuration Guidelines 23-4

Enabling UDLD Globally 23-5

Enabling UDLD on an Interface 23-5

Resetting an Interface Disabled by UDLD 23-6

Displaying UDLD Status 23-6

CHAPTER 24 **Configuring SPAN and RSPAN** 24-1

Understanding SPAN and RSPAN 24-1

Local SPAN 24-2

Remote SPAN 24-2

SPAN and RSPAN Concepts and Terminology 24-3

SPAN Sessions 24-3

Monitored Traffic 24-4

Source Ports 24-5

Source VLANs 24-6

VLAN Filtering 24-6

Destination Port 24-7

RSPAN VLAN 24-8

SPAN and RSPAN Interaction with Other Features 24-8

Configuring SPAN and RSPAN 24-9

Default SPAN and RSPAN Configuration 24-10

Configuring Local SPAN 24-10
SPAN Configuration Guidelines 24-10
Creating a Local SPAN Session 24-11
Creating a Local SPAN Session and Configuring Ingress Traffic 24-13
Specifying VLANs to Filter 24-15
Configuring RSPAN 24-16
RSPAN Configuration Guidelines 24-16
Configuring a VLAN as an RSPAN VLAN 24-17
Creating an RSPAN Source Session 24-17
Creating an RSPAN Destination Session 24-19
Creating an RSPAN Destination Session and Configuring Ingress Traffic 24-20
Specifying VLANs to Filter 24-21
Displaying SPAN and RSPAN Status 24-22

CHAPTER 25 **Configuring RMON 25-1**
Understanding RMON 25-1
Configuring RMON 25-2
Default RMON Configuration 25-3
Configuring RMON Alarms and Events 25-3
Collecting Group History Statistics on an Interface 25-5
Collecting Group Ethernet Statistics on an Interface 25-6
Displaying RMON Status 25-6

CHAPTER 26 **Configuring System Message Logging 26-1**
Understanding System Message Logging 26-1
Configuring System Message Logging 26-2
System Log Message Format 26-2
Default System Message Logging Configuration 26-3
Disabling Message Logging 26-3
Setting the Message Display Destination Device 26-4
Synchronizing Log Messages 26-5
Enabling and Disabling Time Stamps on Log Messages 26-7
Enabling and Disabling Sequence Numbers in Log Messages 26-7
Defining the Message Severity Level 26-8
Limiting Syslog Messages Sent to the History Table and to SNMP 26-9
Configuring UNIX Syslog Servers 26-10
Logging Messages to a UNIX Syslog Daemon 26-10
Configuring the UNIX System Logging Facility 26-11
Displaying the Logging Configuration 26-12

CHAPTER 27 **Configuring SNMP** 27-1

Understanding SNMP 27-1
SNMP Versions 27-2
SNMP Manager Functions 27-3
SNMP Agent Functions 27-4
SNMP Community Strings 27-4
Using SNMP to Access MIB Variables 27-4
SNMP Notifications 27-5
SNMP ifIndex MIB Object Values 27-5
Configuring SNMP 27-6
Default SNMP Configuration 27-6
SNMP Configuration Guidelines 27-6
Disabling the SNMP Agent 27-7
Configuring Community Strings 27-8
Configuring SNMP Groups and Users 27-9
Configuring SNMP Notifications 27-11
Setting the Agent Contact and Location Information 27-14
Limiting TFTP Servers Used Through SNMP 27-15
SNMP Examples 27-15
Displaying SNMP Status 27-16

CHAPTER 28 **Configuring Network Security with ACLs** 28-1

Understanding ACLs 28-1
Supported ACLs 28-2
Port ACLs 28-3
Router ACLs 28-4
VLAN Maps 28-5
Handling Fragmented and Unfragmented Traffic 28-5
Configuring IPv4 ACLs 28-6
Creating Standard and Extended IPv4 ACLs 28-7
IPv4 Access List Numbers 28-8
ACL Logging 28-8
Creating a Numbered Standard ACL 28-9
Creating a Numbered Extended ACL 28-10
Resequencing ACEs in an ACL 28-14
Creating Named Standard and Extended ACLs 28-14
Using Time Ranges with ACLs 28-16
Including Comments in ACLs 28-18
Applying an IPv4 ACL to a Terminal Line 28-18

Applying an IPv4 ACL to an Interface 28-19
Hardware and Software Treatment of IP ACLs 28-21
IPv4 ACL Configuration Examples 28-21
Numbered ACLs 28-23
Extended ACLs 28-23
Named ACLs 28-23
Time Range Applied to an IP ACL 28-24
Commented IP ACL Entries 28-24
ACL Logging 28-25
Creating Named MAC Extended ACLs 28-26
Applying a MAC ACL to a Layer 2 Interface 28-28
Configuring VLAN Maps 28-29
VLAN Map Configuration Guidelines 28-30
Creating a VLAN Map 28-31
Examples of ACLs and VLAN Maps 28-31
Applying a VLAN Map to a VLAN 28-33
Using VLAN Maps in Your Network 28-34
Wiring Closet Configuration 28-34
Denying Access to a Server on Another VLAN 28-35
Using VLAN Maps with Router ACLs 28-36
VLAN Maps and Router ACL Configuration Guidelines 28-36
Examples of Router ACLs and VLAN Maps Applied to VLANs 28-37
ACLs and Switched Packets 28-37
ACLs and Routed Packets 28-38
ACLs and Multicast Packets 28-39
Displaying IPv4 ACL Configuration 28-39

CHAPTER 29 **Configuring Control-Plane Security 29-1**
Understanding Control-Plane Security 29-1
Configuring Control-Plane Security 29-4
Monitoring Control-Plane Security 29-5

CHAPTER 30 **Configuring QoS 30-1**
Understanding QoS 30-1
Modular QoS CLI 30-3
Input and Output Policies 30-4
Input Policy Maps 30-4
Output Policy Maps 30-5

Classification 30-5
Class Maps 30-6
The match Command 30-7
Classification Based on Layer 2 CoS 30-7
Classification Based on IP Precedence 30-8
Classification Based on IP DSCP 30-8
Classification Comparisons 30-9
Classification Based on QoS ACLs 30-10
Classification Based on QoS Groups 30-10
Table Maps 30-11
Policing 30-12
Individual Policing 30-13
Aggregate Policing 30-14
Unconditional Priority Policing 30-15
Marking 30-16
Congestion Management and Scheduling 30-18
Traffic Shaping 30-19
Class-Based Weighted Fair Queuing 30-21
Priority Queuing 30-22
Congestion Avoidance and Queuing 30-24
Configuring QoS 30-26
Default QoS Configuration 30-27
QoS Configuration Guidelines 30-27
Using ACLs to Classify Traffic 30-27
Creating IP Standard ACLs 30-28
Creating IP Extended ACLs 30-29
Creating Layer 2 MAC ACLs 30-30
Using Class Maps to Define a Traffic Class 30-31
Configuring Table Maps 30-33
Attaching a Traffic Policy to an Interface 30-35
Configuring Input Policy Maps 30-35
Configuring Input Policy Maps with Individual Policing 30-36
Configuring Input Policy Maps with Aggregate Policing 30-39
Configuring Input Policy Maps with Marking 30-41
Configuring Output Policy Maps 30-43
Configuring Output Policy Maps with Class-Based-Weighted-Queuing 30-44
Configuring Output Policy Maps with Class-Based Shaping 30-46
Configuring Output Policy Maps with Port Shaping 30-47
Configuring Output Policy Maps with Class-Based Priority Queuing 30-48
Configuring Output Policy Maps with Weighted Tail Drop 30-53

Displaying QoS Information 30-55
QoS Statistics 30-55
Configuration Examples for Policy Maps 30-56
QoS Configuration for Customer A 30-56
QoS Configuration for Customer B 30-58
Modifying Output Policies and Adding or Deleting Classification Criteria 30-59
Modifying Output Policies and Changing Queuing or Scheduling Parameters 30-60
Modifying Output Policies and Adding or Deleting Configured Actions 30-60
Modifying Output Policies and Adding or Deleting a Class 30-61

CHAPTER 31 **Configuring EtherChannels** 31-1
Understanding EtherChannels 31-1
EtherChannel Overview 31-2
Port-Channel Interfaces 31-3
Port Aggregation Protocol 31-4
PAgP Modes 31-5
PAgP Interaction with Other Features 31-5
Link Aggregation Control Protocol 31-6
LACP Modes 31-6
LACP Interaction with Other Features 31-6
EtherChannel On Mode 31-7
Load Balancing and Forwarding Methods 31-7
Configuring EtherChannels 31-9
Default EtherChannel Configuration 31-9
EtherChannel Configuration Guidelines 31-10
Configuring Layer 2 EtherChannels 31-11
Configuring Layer 3 EtherChannels 31-13
Creating Port-Channel Logical Interfaces 31-13
Configuring the Physical Interfaces 31-14
Configuring EtherChannel Load Balancing 31-16
Configuring the PAgP Learn Method and Priority 31-17
Configuring LACP Hot-Standby Ports 31-18
Configuring the LACP System Priority 31-19
Configuring the LACP Port Priority 31-20
Displaying EtherChannel, PAgP, and LACP Status 31-21

CHAPTER 32 **Configuring IP Unicast Routing** 32-1

Understanding IP Routing 32-2
Types of Routing 32-2
Steps for Configuring Routing 32-3
Configuring IP Addressing 32-4
Default Addressing Configuration 32-4
Assigning IP Addresses to Network Interfaces 32-5
Use of Subnet Zero 32-6
Classless Routing 32-6
Configuring Address Resolution Methods 32-7
Define a Static ARP Cache 32-8
Set ARP Encapsulation 32-9
Enable Proxy ARP 32-10
Routing Assistance When IP Routing is Disabled 32-10
Proxy ARP 32-10
Default Gateway 32-11
ICMP Router Discovery Protocol (IRDP) 32-11
Configuring Broadcast Packet Handling 32-12
Enabling Directed Broadcast-to-Physical Broadcast Translation 32-13
Forwarding UDP Broadcast Packets and Protocols 32-14
Establishing an IP Broadcast Address 32-15
Flooding IP Broadcasts 32-15
Monitoring and Maintaining IP Addressing 32-17
Enabling IPv4 Unicast Routing 32-17
Configuring RIP 32-18
Default RIP Configuration 32-19
Configuring Basic RIP Parameters 32-19
Configuring RIP Authentication 32-21
Configuring Summary Addresses and Split Horizon 32-21
Configuring Split Horizon 32-23
Configuring OSPF 32-23
Default OSPF Configuration 32-24
Configuring Basic OSPF Parameters 32-26
Configuring OSPF Interfaces 32-26
Configuring OSPF Area Parameters 32-27
Configuring Other OSPF Parameters 32-29
Changing LSA Group Pacing 32-30
Configuring a Loopback Interface 32-31
Monitoring OSPF 32-31

Configuring EIGRP 32-32
Default EIGRP Configuration 32-34
Configuring Basic EIGRP Parameters 32-35
Configuring EIGRP Interfaces 32-36
Configuring EIGRP Route Authentication 32-37
Monitoring and Maintaining EIGRP 32-38
Configuring BGP 32-38
Default BGP Configuration 32-40
Enabling BGP Routing 32-42
Managing Routing Policy Changes 32-45
Configuring BGP Decision Attributes 32-46
Configuring BGP Filtering with Route Maps 32-48
Configuring BGP Filtering by Neighbor 32-49
Configuring Prefix Lists for BGP Filtering 32-50
Configuring BGP Community Filtering 32-51
Configuring BGP Neighbors and Peer Groups 32-52
Configuring Aggregate Addresses 32-54
Configuring Routing Domain Confederations 32-55
Configuring BGP Route Reflectors 32-56
Configuring Route Dampening 32-57
Monitoring and Maintaining BGP 32-58
Configuring Multi-VRF CE 32-59
Understanding Multi-VRF CE 32-59
Default Multi-VRF CE Configuration 32-61
Multi-VRF CE Configuration Guidelines 32-61
Configuring VRFs 32-62
Configuring a VPN Routing Session 32-63
Configuring BGP PE to CE Routing Sessions 32-64
Multi-VRF CE Configuration Example 32-64
Displaying Multi-VRF CE Status 32-68
Configuring Protocol-Independent Features 32-69
Configuring Cisco Express Forwarding 32-69
Configuring the Number of Equal-Cost Routing Paths 32-70
Configuring Static Unicast Routes 32-71
Specifying Default Routes and Networks 32-72
Using Route Maps to Redistribute Routing Information 32-73
Configuring Policy-Based Routing 32-76
PBR Configuration Guidelines 32-77
Enabling PBR 32-78

Filtering Routing Information 32-79
Setting Passive Interfaces 32-79
Controlling Advertising and Processing in Routing Updates 32-80
Filtering Sources of Routing Information 32-81
Managing Authentication Keys 32-82
Monitoring and Maintaining the IP Network 32-83

CHAPTER 33 **Configuring HSRP 33-1**
Understanding HSRP 33-1
Multiple HSRP 33-3
Configuring HSRP 33-4
Default HSRP Configuration 33-4
HSRP Configuration Guidelines 33-5
Enabling HSRP 33-5
Configuring HSRP Priority 33-6
Configuring MHSRP 33-9
Configuring HSRP Authentication and Timers 33-9
Enabling HSRP Support for ICMP Redirect Messages 33-11
Displaying HSRP Configurations 33-11

CHAPTER 34 **Configuring IP Multicast Routing 34-1**
Understanding Cisco's Implementation of IP Multicast Routing 34-2
Understanding IGMP 34-2
IGMP Version 1 34-3
IGMP Version 2 34-3
Understanding PIM 34-3
PIM Versions 34-3
PIM Modes 34-4
Auto-RP 34-4
Bootstrap Router 34-5
Multicast Forwarding and Reverse Path Check 34-5
Configuring IP Multicast Routing 34-7
Default Multicast Routing Configuration 34-7
Multicast Routing Configuration Guidelines 34-7
PIMv1 and PIMv2 Interoperability 34-8
Auto-RP and BSR Configuration Guidelines 34-8
Configuring Basic Multicast Routing 34-9

Configuring a Rendezvous Point 34-10
Manually Assigning an RP to Multicast Groups 34-11
Configuring Auto-RP 34-12
Configuring PIMv2 BSR 34-16
Using Auto-RP and a BSR 34-20
Monitoring the RP Mapping Information 34-21
Troubleshooting PIMv1 and PIMv2 Interoperability Problems 34-21
Configuring Advanced PIM Features 34-21
Understanding PIM Shared Tree and Source Tree 34-21
Delaying the Use of PIM Shortest-Path Tree 34-23
Modifying the PIM Router-Query Message Interval 34-24
Configuring Optional IGMP Features 34-25
Default IGMP Configuration 34-25
Configuring the Switch as a Member of a Group 34-25
Controlling Access to IP Multicast Groups 34-26
Changing the IGMP Version 34-27
Modifying the IGMP Host-Query Message Interval 34-28
Changing the IGMP Query Timeout for IGMPv2 34-29
Changing the Maximum Query Response Time for IGMPv2 34-29
Configuring the Switch as a Statically Connected Member 34-30
Configuring Optional Multicast Routing Features 34-31
Configuring sdr Listener Support 34-31
Enabling sdr Listener Support 34-31
Limiting How Long an sdr Cache Entry Exists 34-32
Configuring an IP Multicast Boundary 34-32
Monitoring and Maintaining IP Multicast Routing 34-34
Clearing Caches, Tables, and Databases 34-34
Displaying System and Network Statistics 34-34
Monitoring IP Multicast Routing 34-35

CHAPTER 35 **Configuring MSDP** 35-1
Understanding MSDP 35-1
MSDP Operation 35-2
MSDP Benefits 35-3
Configuring MSDP 35-4
Default MSDP Configuration 35-4
Configuring a Default MSDP Peer 35-4
Caching Source-Active State 35-6
Requesting Source Information from an MSDP Peer 35-8

Controlling Source Information that Your Switch Originates 35-8
Redistributing Sources 35-9
Filtering Source-Active Request Messages 35-11
Controlling Source Information that Your Switch Forwards 35-12
Using a Filter 35-12
Using TTL to Limit the Multicast Data Sent in SA Messages 35-14
Controlling Source Information that Your Switch Receives 35-14
Configuring an MSDP Mesh Group 35-16
Shutting Down an MSDP Peer 35-16
Including a Bordering PIM Dense-Mode Region in MSDP 35-17
Configuring an Originating Address other than the RP Address 35-18
Monitoring and Maintaining MSDP 35-19

CHAPTER 36 **Troubleshooting** 36-1
Recovering from Corrupted Software By Using the Xmodem Protocol 36-2
Recovering from a Lost or Forgotten Password 36-3
Procedure with Password Recovery Enabled 36-5
Procedure with Password Recovery Disabled 36-7
Preventing Autonegotiation Mismatches 36-8
SFP Module Security and Identification 36-9
Monitoring SFP Module Status 36-9
Monitoring Temperature 36-9
Using Ping 36-10
Understanding Ping 36-10
Using Ping 36-10
All Software Versions 36-11
Metro IP Access Image 36-11
Ping Responses 36-12
Summary 36-13
Using Layer 2 Traceroute 36-13
Understanding Layer 2 Traceroute 36-13
Layer 2 Traceroute Usage Guidelines 36-14
Displaying the Physical Path 36-15
Using IP Traceroute 36-15
Understanding IP Traceroute 36-15
Executing IP Traceroute 36-16

Using TDR 36-17
Understanding TDR 36-17
Running TDR and Displaying the Results 36-17
Using Debug Commands 36-18
Enabling Debugging on a Specific Feature 36-18
Enabling All-System Diagnostics 36-19
Redirecting Debug and Error Message Output 36-19
Using the show platform forward Command 36-19
Using the crashinfo File 36-22

APPENDIX A
Supported MIBs A-1
MIB List A-1
Using FTP to Access the MIB Files A-3

APPENDIX B
Working with the Cisco IOS File System, Configuration Files, and Software Images B-1
Working with the Flash File System B-1
Displaying Available File Systems B-2
Setting the Default File System B-3
Displaying Information about Files on a File System B-3
Changing Directories and Displaying the Working Directory B-3
Creating and Removing Directories B-4
Copying Files B-4
Deleting Files B-5
Creating, Displaying, and Extracting tar Files B-5
Creating a tar File B-6
Displaying the Contents of a tar File B-6
Extracting a tar File B-7
Displaying the Contents of a File B-7
Working with Configuration Files B-8
Guidelines for Creating and Using Configuration Files B-8
Configuration File Types and Location B-9
Creating a Configuration File By Using a Text Editor B-9
Copying Configuration Files By Using TFTP B-10
Preparing to Download or Upload a Configuration File By Using TFTP B-10
Downloading the Configuration File By Using TFTP B-10
Uploading the Configuration File By Using TFTP B-11

Copying Configuration Files By Using FTP B-11
Preparing to Download or Upload a Configuration File By Using FTP B-12
Downloading a Configuration File By Using FTP B-13
Uploading a Configuration File By Using FTP B-14
Copying Configuration Files By Using RCP B-15
Preparing to Download or Upload a Configuration File By Using RCP B-15
Downloading a Configuration File By Using RCP B-16
Uploading a Configuration File By Using RCP B-17
Clearing Configuration Information B-18
Clearing the Startup Configuration File B-18
Deleting a Stored Configuration File B-18
Working with Software Images B-18
Image Location on the Switch B-19
tar File Format of Images on a Server or Cisco.com B-19
Copying Image Files By Using TFTP B-20
Preparing to Download or Upload an Image File By Using TFTP B-21
Downloading an Image File By Using TFTP B-21
Uploading an Image File By Using TFTP B-23
Copying Image Files By Using FTP B-23
Preparing to Download or Upload an Image File By Using FTP B-24
Downloading an Image File By Using FTP B-25
Uploading an Image File By Using FTP B-27
Copying Image Files By Using RCP B-28
Preparing to Download or Upload an Image File By Using RCP B-28
Downloading an Image File By Using RCP B-29
Uploading an Image File By Using RCP B-31

APPENDIX C **Unsupported Commands in Cisco IOS Release 12.2(25)EX** C-1
Access Control Lists C-1
Unsupported Privileged EXEC Commands C-1
Unsupported Global Configuration Commands C-1
ARP Commands C-1
Unsupported Global Configuration Commands C-1
Unsupported Interface Configuration Commands C-2
Unsupported Debug Commands C-2
HSRP C-2
Unsupported Global Configuration Commands C-2
Unsupported Interface Configuration Commands C-2

IGMP Snooping Commands C-2
Unsupported Global Configuration Commands C-2
Interface Commands C-3
Unsupported Privileged EXEC Commands C-3
Unsupported Global Configuration Commands C-3
Unsupported Interface Configuration Commands C-3
IP Multicast Routing C-3
Unsupported Privileged EXEC Commands C-3
Unsupported Global Configuration Commands C-4
Unsupported Interface Configuration Commands C-4
IP Unicast Routing C-4
Unsupported Privileged EXEC or User EXEC Commands C-4
Unsupported Global Configuration Commands C-5
Unsupported Interface Configuration Commands C-5
Unsupported BGP Router Configuration Commands C-6
Unsupported VPN Configuration Commands C-6
Unsupported Route Map Commands C-6
MAC Address Commands C-7
Unsupported Privileged EXEC Commands C-7
Unsupported Global Configuration Commands C-7
Miscellaneous C-7
Unsupported Global Configuration Commands C-7
Unsupported Privileged EXEC Commands C-8
Unsupported show platform Commands C-8
MSDP C-8
Unsupported Privileged EXEC Commands C-8
Unsupported Global Configuration Commands C-8
NetFlow Commands C-8
Unsupported Global Configuration Commands C-8
QoS C-9
Unsupported Global Configuration Commands C-9
Unsupported Interface Configuration Commands C-9
RADIUS C-9
Unsupported Global Configuration Commands C-9
SNMP C-9
Unsupported Global Configuration Commands C-9

Spanning Tree C-9
Unsupported Global Configuration Command C-9
Unsupported Interface Configuration Command C-9
VLAN C-10
Unsupported Global Configuration Commands C-10
Unsupported User EXEC Commands C-10

INDEX



Preface

Audience

This guide is for the networking professional managing the Cisco Metro Ethernet (ME) 3400 Series Ethernet Access switch, hereafter referred to as the *switch*. Before using this guide, you should have experience working with the Cisco IOS software and be familiar with the concepts and terminology of Ethernet and local area networking.

Purpose

This guide provides procedures for using the commands that have been created or changed for use with the Cisco ME 3400 switch. It does not provide detailed information about these commands. For detailed information about these commands, see the *Cisco ME 3400 Ethernet Access Switch Command Reference* for this release. For information about the standard Cisco IOS Release 12.2 commands, see the Cisco IOS documentation set available from the Cisco.com home page at **Service and Support > Technical Documents**. On the Cisco Product Documentation home page, select **Release 12.2** from the Cisco IOS Software drop-down list.

This guide does not describe system messages you might encounter or how to install your switch. For more information, see the *Cisco ME 3400 Ethernet Access Switch System Message Guide* for this release and the *Cisco ME 3400 Ethernet Access Switch Hardware Installation Guide*.

For the latest documentation updates, see the release notes for this release.

Conventions

This publication uses these conventions to convey instructions and information:

Command descriptions use these conventions:

- Commands and keywords are in **boldface** text.
- Arguments for which you supply values are in *italic*.
- Square brackets ([]) mean optional elements.
- Braces ({ }) group required choices, and vertical bars (|) separate the alternative elements.
- Braces and vertical bars within square brackets ([{ | }]) mean a required choice within an optional element.

Interactive examples use these conventions:

- Terminal sessions and system displays are in `screen` font.
- Information you enter is in **`boldface screen`** font.
- Nonprinting characters, such as passwords or tabs, are in angle brackets (< >).

Notes, cautions, and timesavers use these conventions and symbols:

Note Means *reader take note*. Notes contain helpful suggestions or references to materials not contained in this manual.

Caution Means *reader be careful*. In this situation, you might do something that could result in equipment damage or loss of data.

Related Publications

These documents provide complete information about the switch and are available from this Cisco.com site:

http://www.cisco.com/univercd/cc/td/doc/product/metro/me3400/index.htm

Note Before installing, configuring, or upgrading the switch, see these documents:

- For initial configuration information, see the "Configuring the Switch with the CLI-Based Setup Program" appendix in the hardware installation guide.
- For upgrading information, see the "Downloading Software" section in the release notes.

You can order printed copies of documents with a DOC-xxxxxx= number from the Cisco.com sites and from the telephone numbers listed in the "Obtaining Documentation" section on page xxxv.

- *Release Notes for the Cisco ME 3400 Ethernet Access Switch* (not orderable but available on Cisco.com)
- *Cisco ME 3400 Ethernet Access Switch Software Configuration Guide* (order number DOC-7817058=)
- *Cisco ME 3400 Ethernet Access Switch Command Reference* (order number DOC-7817060=)
- *Cisco ME 3400 Ethernet Access Switch System Message Guide* (order number DOC-7817062=)
- *Cisco ME 3400 Ethernet Access Switch Hardware Installation Guide* (not orderable but available on Cisco.com)
- *Cisco ME 3400 and ME 2400 Ethernet Access Switches Getting Started Guide (*order number DOC-7817050=)
- *Regulatory Compliance and Safety Information for the Cisco ME 3400 and ME 2400 Ethernet Access Switches* (order number DOC-7817051)
- *Cisco Small Form-Factor Pluggable Modules Installation Notes* (order number DOC-7815160=)
- *Cisco CWDM GBIC and CWDM SFP Installation Note* (not orderable but available on Cisco.com)

- *Cisco Gigabit Ethernet Transceiver Modules Compatibility Matrix* (not orderable but available on Cisco.com)
- *Cisco 100-Megabit Ethernet SFP Modules Compatibility Matrix* (not orderable but available on Cisco.com)
- *Cisco CWDM SFP Transceiver Compatibility Matrix* (not orderable but available on Cisco.com)

Obtaining Documentation

Cisco documentation and additional literature are available on Cisco.com. Cisco also provides several ways to obtain technical assistance and other technical resources. These sections explain how to obtain technical information from Cisco Systems.

Cisco.com

You can access the most current Cisco documentation at this URL:

http://www.cisco.com/techsupport

You can access the Cisco website at this URL:

http://www.cisco.com

You can access international Cisco websites at this URL:

http://www.cisco.com/public/countries_languages.shtml

Product Documentation DVD

Cisco documentation and additional literature are available in the Product Documentation DVD package, which may have shipped with your product. The Product Documentation DVD is updated regularly and may be more current than printed documentation.

The Product Documentation DVD is a comprehensive library of technical product documentation on portable media. The DVD enables you to access multiple versions of hardware and software installation, configuration, and command guides for Cisco products and to view technical documentation in HTML. With the DVD, you have access to the same documentation that is found on the Cisco website without being connected to the Internet. Certain products also have .pdf versions of the documentation available.

The Product Documentation DVD is available as a single unit or as a subscription. Registered Cisco.com users (Cisco direct customers) can order a Product Documentation DVD (product number DOC-DOCDVD=) from Cisco Marketplace at this URL:

http://www.cisco.com/go/marketplace/

Ordering Documentation

Beginning June 30, 2005, registered Cisco.com users may order Cisco documentation at the Product Documentation Store in the Cisco Marketplace at this URL:

http://www.cisco.com/go/marketplace/

Nonregistered Cisco.com users can order technical documentation from 8:00 a.m. to 5:00 p.m. (0800 to 1700) PDT by calling 1 866 463-3487 in the United States and Canada, or elsewhere by calling 011 408 519-5055. You can also order documentation by e-mail at tech-doc-store-mkpl@external.cisco.com or by fax at 1 408 519-5001 in the United States and Canada, or elsewhere at 011 408 519-5001.

Documentation Feedback

You can rate and provide feedback about Cisco technical documents by completing the online feedback form that appears with the technical documents on Cisco.com.

You can send comments about Cisco documentation to bug-doc@cisco.com.

You can submit comments by using the response card (if present) behind the front cover of your document or by writing to the following address:

Cisco Systems
Attn: Customer Document Ordering
170 West Tasman Drive
San Jose, CA 95134-9883

We appreciate your comments.

Cisco Product Security Overview

Cisco provides a free online Security Vulnerability Policy portal at this URL:

http://www.cisco.com/en/US/products/products_security_vulnerability_policy.html

From this site, you can perform these tasks:

- Report security vulnerabilities in Cisco products.
- Obtain assistance with security incidents that involve Cisco products.
- Register to receive security information from Cisco.

A current list of security advisories and notices for Cisco products is available at this URL:

http://www.cisco.com/go/psirt

If you prefer to see advisories and notices as they are updated in real time, you can access a Product Security Incident Response Team Really Simple Syndication (PSIRT RSS) feed from this URL:

http://www.cisco.com/en/US/products/products_psirt_rss_feed.html

Reporting Security Problems in Cisco Products

Cisco is committed to delivering secure products. We test our products internally before we release them, and we strive to correct all vulnerabilities quickly. If you think that you might have identified a vulnerability in a Cisco product, contact PSIRT:

- Emergencies—security-alert@cisco.com

 An emergency is either a condition in which a system is under active attack or a condition for which a severe and urgent security vulnerability should be reported. All other conditions are considered nonemergencies.

- Nonemergencies—psirt@cisco.com

In an emergency, you can also reach PSIRT by telephone:

- 1 877 228-7302
- 1 408 525-6532



Tip We encourage you to use Pretty Good Privacy (PGP) or a compatible product to encrypt any sensitive information that you send to Cisco. PSIRT can work from encrypted information that is compatible with PGP versions 2.*x* through 8.*x*.

Never use a revoked or an expired encryption key. The correct public key to use in your correspondence with PSIRT is the one linked in the Contact Summary section of the Security Vulnerability Policy page at this URL:

http://www.cisco.com/en/US/products/products_security_vulnerability_policy.html

The link on this page has the current PGP key ID in use.

Obtaining Technical Assistance

Cisco Technical Support provides 24-hour-a-day award-winning technical assistance. The Cisco Technical Support & Documentation website on Cisco.com features extensive online support resources. In addition, if you have a valid Cisco service contract, Cisco Technical Assistance Center (TAC) engineers provide telephone support. If you do not have a valid Cisco service contract, contact your reseller.

Cisco Technical Support & Documentation Website

The Cisco Technical Support & Documentation website provides online documents and tools for troubleshooting and resolving technical issues with Cisco products and technologies. The website is available 24 hours a day, at this URL:

http://www.cisco.com/techsupport

Access to all tools on the Cisco Technical Support & Documentation website requires a Cisco.com user ID and password. If you have a valid service contract but do not have a user ID or password, you can register at this URL:

http://tools.cisco.com/RPF/register/register.do

Note Use the Cisco Product Identification (CPI) tool to locate your product serial number before submitting a web or phone request for service. You can access the CPI tool from the Cisco Technical Support & Documentation website by clicking the **Tools & Resources** link under Documentation & Tools. Choose **Cisco Product Identification Tool** from the Alphabetical Index drop-down list, or click the **Cisco Product Identification Tool** link under Alerts & RMAs. The CPI tool offers three search options: by product ID or model name; by tree view; or for certain products, by copying and pasting **show** command output. Search results show an illustration of your product with the serial number label location highlighted. Locate the serial number label on your product and record the information before placing a service call.

Submitting a Service Request

Using the online TAC Service Request Tool is the fastest way to open S3 and S4 service requests. (S3 and S4 service requests are those in which your network is minimally impaired or for which you require product information.) After you describe your situation, the TAC Service Request Tool provides recommended solutions. If your issue is not resolved using the recommended resources, your service request is assigned to a Cisco engineer. The TAC Service Request Tool is located at this URL:

http://www.cisco.com/techsupport/servicerequest

For S1 or S2 service requests or if you do not have Internet access, contact the Cisco TAC by telephone. (S1 or S2 service requests are those in which your production network is down or severely degraded.) Cisco engineers are assigned immediately to S1 and S2 service requests to help keep your business operations running smoothly.

To open a service request by telephone, use one of the following numbers:

Asia-Pacific: +61 2 8446 7411 (Australia: 1 800 805 227)
EMEA: +32 2 704 55 55
USA: 1 800 553-2447

For a complete list of Cisco TAC contacts, go to this URL:

http://www.cisco.com/techsupport/contacts

Definitions of Service Request Severity

To ensure that all service requests are reported in a standard format, Cisco has established severity definitions.

Severity 1 (S1)—Your network is “down,” or there is a critical impact to your business operations. You and Cisco will commit all necessary resources around the clock to resolve the situation.

Severity 2 (S2)—Operation of an existing network is severely degraded, or significant aspects of your business operation are negatively affected by inadequate performance of Cisco products. You and Cisco will commit full-time resources during normal business hours to resolve the situation.

Severity 3 (S3)—Operational performance of your network is impaired, but most business operations remain functional. You and Cisco will commit resources during normal business hours to restore service to satisfactory levels.

Severity 4 (S4)—You require information or assistance with Cisco product capabilities, installation, or configuration. There is little or no effect on your business operations.

Obtaining Additional Publications and Information

Information about Cisco products, technologies, and network solutions is available from various online and printed sources.

- Cisco Marketplace provides a variety of Cisco books, reference guides, documentation, and logo merchandise. Visit Cisco Marketplace, the company store, at this URL:

 http://www.cisco.com/go/marketplace/

- *Cisco Press* publishes a wide range of general networking, training and certification titles. Both new and experienced users will benefit from these publications. For current Cisco Press titles and other information, go to Cisco Press at this URL:

 http://www.ciscopress.com

- *Packet* magazine is the Cisco Systems technical user magazine for maximizing Internet and networking investments. Each quarter, Packet delivers coverage of the latest industry trends, technology breakthroughs, and Cisco products and solutions, as well as network deployment and troubleshooting tips, configuration examples, customer case studies, certification and training information, and links to scores of in-depth online resources. You can access Packet magazine at this URL:

 http://www.cisco.com/packet

- *iQ Magazine* is the quarterly publication from Cisco Systems designed to help growing companies learn how they can use technology to increase revenue, streamline their business, and expand services. The publication identifies the challenges facing these companies and the technologies to help solve them, using real-world case studies and business strategies to help readers make sound technology investment decisions. You can access iQ Magazine at this URL:

 http://www.cisco.com/go/iqmagazine

 or view the digital edition at this URL:

 http://ciscoiq.texterity.com/ciscoiq/sample/

- *Internet Protocol Journal* is a quarterly journal published by Cisco Systems for engineering professionals involved in designing, developing, and operating public and private internets and intranets. You can access the Internet Protocol Journal at this URL:

 http://www.cisco.com/ipj

- Networking products offered by Cisco Systems, as well as customer support services, can be obtained at this URL:

 http://www.cisco.com/en/US/products/index.html

- Networking Professionals Connection is an interactive website for networking professionals to share questions, suggestions, and information about networking products and technologies with Cisco experts and other networking professionals. Join a discussion at this URL:

 http://www.cisco.com/discuss/networking

- World-class networking training is available from Cisco. You can view current offerings at this URL:

 http://www.cisco.com/en/US/learning/index.html



Overview

This chapter provides these topics about the Cisco Metro Ethernet (ME) 3400 Series Ethernet Access switch software:

- Features, page 1-1
- Default Settings After Initial Switch Configuration, page 1-8
- Network Configuration Examples, page 1-11
- Where to Go Next, page 1-15

In this document, IP refers to IP Version 4 (IPv4).

Features

The switch ships with one of these software images installed:

- The metro base image provides basic Metro Ethernet features.
- The metro access image includes additional features such as IEEE 802.1Q tunneling, Layer 2 protocol tunneling, dynamic ARP inspection, and IP source guard.
- The metro IP access image adds Layer 3 functionality such as IP routing support for Routing Information Protocol (RIP), Open Shortest Path First (OSPF) Protocol, Border Gateway Protocol (BGP), and Enhanced Interior Gateway Routing Protocol (EIGRP), multiple VPN routing/forwarding on customer edge devices, (multi-VRF-CE), and IP multicast routing Protocol-Independent Multicast (PIM) sparse mode (SM) and dense mode (DM).

Note Unless otherwise noted, all features described in this chapter and in this guide are supported on all images.

Some features noted in this chapter are available only on the cryptographic (that is, supports encryption) versions of the switch software image. You must obtain authorization to use this feature and to download the cryptographic version of the software from Cisco.com. For more information, see the release notes for this release.

The Cisco ME switch has two different types of interfaces: network node interfaces (NNIs) to connect to the service provider network and user network interfaces (UNIs) to connect to customer networks. Some features are supported only on one of these port types.

The switch has these features:

- Performance Features, page 1-2
- Management Options, page 1-3
- Manageability Features, page 1-3 (includes a feature requiring the cryptographic versions of the software)
- Availability Features, page 1-4
- VLAN Features, page 1-5
- Security Features, page 1-5 (includes a feature requiring the cryptographic versions of the switch software)
- Quality of Service and Class of Service Features, page 1-6
- Layer 2 Virtual Private Network Services, page 1-7
- Layer 3 Features, page 1-7 (requires metro IP access image)
- Layer 3 VPN Services, page 1-8 (requires metro IP access image)
- Monitoring Features, page 1-8

Performance Features

- Autosensing of port speed and autonegotiation of duplex mode on all switch ports for optimizing bandwidth
- Automatic-medium-dependent interface crossover (auto-MDIX) capability on 10/100 and 10/100/1000 Mbps interfaces and on 10/100/1000 BASE-T/TX small form-factor pluggable (SFP) module interfaces that enables the interface to automatically detect the required cable connection type (straight-through or crossover) and to configure the connection appropriately
- Support for routed frames up to 1546 bytes, for frames up to 9000 bytes that are bridged in hardware, and for frames up to 2000 bytes that are bridged by software.
- IEEE 802.3x flow control on all ports (the switch does not send pause frames)
- EtherChannel for enhanced fault tolerance and for providing up to 2 Gbps (Gigabit EtherChannel) or 800 Mbps (Fast EtherChannel) full duplex of bandwidth between switches, routers, and servers
- Port Aggregation Protocol (PAgP) and Link Aggregation Control Protocol (LACP) for automatic creation of EtherChannel links (supported only on NNIs)
- Forwarding of Layer 2 and Layer 3 packets at Gigabit line rate
- Per-port storm control for preventing broadcast, multicast, and unicast storms
- Port blocking on forwarding unknown Layer 2 unknown unicast, multicast, and bridged broadcast traffic
- Internet Group Management Protocol (IGMP) snooping for IGMP versions 1, 2, and 3 for efficiently forwarding multimedia and multicast traffic
- IGMP report suppression for sending only one IGMP report per multicast router query to the multicast devices (supported only for IGMPv1 or IGMPv2 queries)
- IGMP snooping querier support to configure switch to generate periodic IGMP General Query messages
- Multicast VLAN registration (MVR) to continuously send multicast streams in a multicast VLAN while isolating the streams from subscriber VLANs for bandwidth and security reasons

- IGMP filtering for controlling the set of multicast groups to which hosts on a switch port can belong
- IGMP throttling for configuring the action when the maximum number of entries is in the IGMP forwarding table
- IGMP configurable leave timer to configure the leave latency for the network.
- Switch Database Management (SDM) templates for allocating system resources to maximize support for user-selected features

Management Options

- CLI—The Cisco IOS software supports desktop- and multilayer-switching features. You can access the CLI either by connecting your management station directly to the switch console port or by using Telnet from a remote management station. For more information about the CLI, see Chapter 2, "Using the Command-Line Interface."
- Cisco Configuration Engine—The Cisco Configuration Engine is a network management device that works with embedded Cisco IOS CNS Agents in the switch software. You can automate initial configurations and configuration updates by generating switch-specific configuration changes, sending them to the switch, executing the configuration change, and logging the results. For more information about using Cisco IOS agents, see Chapter 4, "Configuring Cisco IOS CNS Agents."
- SNMP—SNMP management applications such as CiscoWorks2000 LAN Management Suite (LMS) and HP OpenView. You can manage from an SNMP-compatible management station that is running platforms such as HP OpenView or SunNet Manager. The switch supports a comprehensive set of MIB extensions and four remote monitoring (RMON) groups. For more information about using SNMP, see Chapter 27, "Configuring SNMP."

Manageability Features

Note The encrypted Secure Shell (SSH) feature listed in this section is available only on the cryptographic versions of the switch software image.

- Support for DHCP for configuration of switch information (such as IP address, default gateway, hostname, and Domain Name System [DNS] and TFTP server names)
- DHCP relay for forwarding User Datagram Protocol (UDP) broadcasts, including IP address requests, from DHCP clients
- Directed unicast requests to a DNS server for identifying a switch through its IP address and its corresponding hostname and to a TFTP server for administering software upgrades from a TFTP server
- Address Resolution Protocol (ARP) for identifying a switch through its IP address and its corresponding MAC address
- Unicast MAC address filtering to drop packets with specific source or destination MAC addresses
- Configurable MAC address scaling that allows disabling MAC address learning on a VLAN to limit the size of the MAC address table
- Cisco Discovery Protocol (CDP) Versions 1 and 2 for network topology discovery and mapping between the switch and other Cisco devices on the network (supported only on NNIs)
- Network Time Protocol (NTP) for providing a consistent time stamp to all switches from an external source

- Cisco IOS File System (IFS) for providing a single interface to all file systems that the switch uses
- In-band management access for up to 16 simultaneous Telnet connections for multiple CLI-based sessions over the network
- In-band management access for up to five simultaneous, encrypted Secure Shell (SSH) connections for multiple CLI-based sessions over the network (requires the cryptographic versions of the switch software).
- In-band management access through SNMP Versions 1, 2c, and 3 get and set requests
- Out-of-band management access through the switch console port to a directly attached terminal or to a remote terminal through a serial connection or a modem
- User-defined command macros for creating custom switch configurations for simplified deployment across multiple switches

Availability Features

- UniDirectional Link Detection (UDLD) and aggressive UDLD for detecting and disabling unidirectional links on fiber-optic interfaces caused by incorrect fiber-optic wiring or port faults
- IEEE 802.1D Spanning Tree Protocol (STP) for redundant backbone connections and loop-free networks (supported only on NNIs). STP has these features:
 - Up to 128 supported spanning-tree instances
 - Per-VLAN spanning-tree plus (PVST+) for balancing load across VLANs
 - Rapid PVST+ for balancing load across VLANs and providing rapid convergence of spanning-tree instances
- IEEE 802.1s Multiple Spanning Tree Protocol (MSTP) on NNIs for grouping VLANs into a spanning-tree instance and for providing multiple forwarding paths for data traffic and load balancing and rapid per-VLAN Spanning-Tree plus (rapid-PVST+) based on the IEEE 802.1w Rapid Spanning Tree Protocol (RSTP) for rapid convergence of the spanning tree by immediately transitioning root and designated port NNIs to the forwarding state
- Optional spanning-tree features available in PVST+, rapid-PVST+, and MSTP modes on NNIs:
 - Port Fast for eliminating the forwarding delay by enabling an NNI to immediately transition from the blocking state to the forwarding state
 - Bridge protocol data unit (BPDU) guard for shutting down Port Fast-enabled NNIs that receive BPDUs
 - BPDU filtering for preventing a Port Fast-enabled NNI from sending or receiving BPDUs
 - Root guard for preventing switches outside the network core from becoming the spanning-tree root
 - Loop guard for preventing alternate or root port NNIs from becoming designated ports because of a failure that leads to a unidirectional link
- Flex Link Layer 2 interfaces to back up one another as an alternative to STP for basic link redundancy in a nonloop network (requires metro IP access or metro access image)
- HSRP for Layer 3 router redundancy (requires metro IP access image)
- Equal-cost routing for link-level and switch-level redundancy (requires metro IP access image)

VLAN Features

- Support for up to 1005 VLANs for assigning users to VLANs associated with appropriate network resources, traffic patterns, and bandwidth
- Support for VLAN IDs in the full 1 to 4094 range allowed by the IEEE 802.1Q standard
- VLAN Query Protocol (VQP) for dynamic VLAN membership
- IEEE 802.1Q trunking encapsulation on all ports for network moves, adds, and changes; management and control of broadcast and multicast traffic; and network security by establishing VLAN groups for high-security users and network resources
- VLAN 1 minimization for reducing the risk of spanning-tree loops or storms by allowing VLAN 1 to be disabled on any individual VLAN trunk link. With this feature enabled, no user traffic is sent or received on the trunk. The switch CPU continues to send and receive control protocol frames.
- UNI-isolated VLANs to isolate customer VLANs from VLANs of other customers on the same switch. Local switching does not occur among UNIs on the switch that belong to the same UNI isolated VLAN.
- Private VLANs to address VLAN scalability problems, to provide a more controlled IP address allocation, and to allow Layer 2 ports to be isolated from ports on other switches

Security Features

The switch provides security for the subscriber, the switch, and the network.

Subscriber Security

- By default, local switching is disabled among subscriber ports to ensure that subscribers are isolated.
- DHCP snooping to filter untrusted DHCP messages between untrusted hosts and DHCP servers
- IP source guard to restrict traffic on nonrouted interfaces by filtering traffic based on the DHCP snooping database and IP source bindings
- Dynamic ARP inspection to prevent malicious attacks on the switch by not relaying invalid ARP requests and responses to other ports in the same VLAN

Note IP source guard and dynamic ARP inspection are available only when the switch is running the metro IP access or metro access image.

Switch Security

Note The Kerberos feature listed in this section is only available on the cryptographic versions of the switch software.

- Password-protected access (read-only and read-write access) to management interfaces for protection against unauthorized configuration changes

- Configuration file security so that only authenticated and authorized users have access to the configuration file, preventing users from accessing the configuration file by using the password recovery process
- Multilevel security for a choice of security level, notification, and resulting actions
- Port security option for limiting and identifying MAC addresses of the stations allowed to access the port
- Port security aging to set the aging time for secure addresses on a port
- UNI default port state is disabled
- Automatic control-plane protection to protect the CPU from accidental or malicious overload due to Layer 2 control traffic on UNIs
- TACACS+, a proprietary feature for managing network security through a TACACS server
- RADIUS for verifying the identity of, granting access to, and tracking the actions of remote users through authentication, authorization, and accounting (AAA) services
- Kerberos security system to authenticate requests for network resources by using a trusted third party (requires the cryptographic versions of the switch software)

Network Security

- Static MAC addressing for ensuring security
- Standard and extended IP access control lists (ACLs) for defining security policies in both directions on routed interfaces (router ACLs) and VLANs and inbound on Layer 2 interfaces (port ACLs)
- Extended MAC access control lists for defining security policies in the inbound direction on Layer 2 interfaces
- VLAN ACLs (VLAN maps) for providing intra-VLAN security by filtering traffic based on information in the MAC, IP, and TCP/UDP headers
- Source and destination MAC-based ACLs for filtering non-IP traffic
- IEEE 802.1x port-based authentication to prevent unauthorized devices (clients) from gaining access to the network. These features are supported:
 - VLAN assignment for restricting IEEE 802.1x-authenticated users to a specified VLAN
 - Port security for controlling access to IEEE 802.1x ports
 - IEEE 802.1x accounting to track network usage

Quality of Service and Class of Service Features

- Cisco modular quality of service (QoS) command-line (MQC) implementation
- Classification based on IP precedence, Differentiated Services Code Point (DSCP), and IEEE 802.1p class of service (CoS) packet fields, ACL lookup, or assigning a QoS label for output classification
- Policing
 - One-rate policing based on average rate and burst rate for a policer
 - Two-color policing that allows different actions for packets that conform to or exceed the rate
 - Aggregate policing for policers shared by multiple traffic classes

- Weighted tail drop (WTD) as the congestion-avoidance mechanism for managing the queue lengths and providing drop precedences for different traffic classifications
- Table maps for mapping DSCP, CoS, and IP precedence values
- Queuing and Scheduling
 - Shaped round robin (SRR) traffic shaping to mix packets from all queues to minimize traffic burst
 - Class-based traffic shaping to specify a maximum permitted average rate for a traffic class
 - Port shaping to specify the maximum permitted average rate for a port
 - Class-based weighted queuing (CBWFQ) to control bandwidth to a traffic class
 - WTD to adjust queue size for a specified traffic class
 - Low-latency priority queuing to allow preferential treatment to certain traffic

Layer 2 Virtual Private Network Services

Layer 2 virtual private network (VPN) features are only available when the switch is running the metro IP access or metro access image.

- IEEE 802.1Q tunneling enables service providers to offer multiple point Layer 2 VPN services to customers
- Layer 2 protocol tunneling to enable customers to control protocols such as BPDU, CDP, VTP, PAgP, LACP, and UDLD protocols to be tunneled across service-provider networks.

Layer 3 Features

Layer 3 features are only available when the switch is running the metro IP access image.

- HSRP for Layer 3 router redundancy
- IP routing protocols for load balancing and for constructing scalable, routed backbones:
 - RIP Versions 1 and 2
 - OSPF
 - EIGRP
 - BGP Version 4
- IP routing between VLANs (inter-VLAN routing) for full Layer 3 routing between two or more VLANs, allowing each VLAN to maintain its own autonomous data-link domain
- Policy-based routing (PBR) for configuring defined policies for traffic flows
- Static IP routing for manually building a routing table of network path information
- Equal-cost routing for load balancing and redundancy
- Internet Control Message Protocol (ICMP) and ICMP Router Discovery Protocol (IRDP) for using router advertisement and router solicitation messages to discover the addresses of routers on directly attached subnets

- Protocol-Independent Multicast (PIM) for multicast routing within the network, allowing for devices in the network to receive the multicast feed requested and for switches not participating in the multicast to be pruned. Includes support for PIM sparse mode (PIM-SM), PIM dense mode (PIM-DM), and PIM sparse-dense mode
- Multicast Source Discovery Protocol (MSDP) for connecting multiple PIM-SM domains
- DHCP relay for forwarding UDP broadcasts, including IP address requests, from DHCP clients

Layer 3 VPN Services

These features are available only when the switch is running the metro IP access image.

- Multiple VPN routing/forwarding (multi-VRF) instances in customer edge devices (multi-VRF CE) to allow service providers to support multiple virtual private networks (VPNs) and overlap IP addresses between VPNs
- VRF and EIGRP compatibility

Monitoring Features

- Switch LEDs that provide port- and switch-level status
- MAC address notification traps and RADIUS accounting for tracking users on a network by storing the MAC addresses that the switch has learned or removed
- Switched Port Analyzer (SPAN) and Remote SPAN (RSPAN) for traffic monitoring on any port or VLAN
- SPAN and RSPAN support of Intrusion Detection Systems (IDS) to monitor, repel, and report network security violations
- Four groups (history, statistics, alarms, and events) of embedded RMON agents for network monitoring and traffic analysis
- Syslog facility for logging system messages about authentication or authorization errors, resource issues, and time-out events
- Layer 2 traceroute to identify the physical path that a packet takes from a source device to a destination device
- Time Domain Reflector (TDR) to diagnose and resolve cabling problems on copper Ethernet 10/100 ports
- SFP module diagnostic management interface to monitor physical or operational status of an SFP module

Default Settings After Initial Switch Configuration

The switch is designed for plug-and-play operation; you only need to assign basic IP information to the switch and connect it to the other devices in your network. If you have specific network needs, you can change the interface-specific and system-wide settings.

Note For information about assigning an IP address by using the CLI-based setup program, see the hardware installation guide.

If you do not configure the switch at all, the Cisc0 ME 3400 switch operates with the default settings shown in Table 1-1.

Table 1-1 Default Settings After Initial Switch Configuration

Feature	Default Setting	More information in...
Switch IP address, subnet mask, and default gateway	0.0.0.0	Chapter 3, "Assigning the Switch IP Address and Default Gateway"
Domain name	None	
Passwords	None defined	Chapter 5, "Administering the Switch"
TACACS+	Disabled	
RADIUS	Disabled	
System name and prompt	*Switch*	
NTP	Enabled	
DNS	Enabled	
IEEE 802.1x	Disabled	Chapter 8, "Configuring IEEE 802.1x Port-Based Authentication"
DHCP		
• DHCP client	Enabled	Chapter 3, "Assigning the Switch IP Address and Default Gateway" Chapter 18, "Configuring DHCP Features and IP Source Guard"
• DHCP relay agent	Enabled (if the device is acting as a DHCP relay agent and is configured and enabled)	
Port parameters		
• Operating mode	Layer 2 (switchport)	Chapter 9, "Configuring Interface Characteristics"
• Port enable state	Enabled NNIs; disabled UNIs	
• Interface speed and duplex mode	Autonegotiate	
• Auto-MDIX	Enabled	
• Flow control	Off	
Command Macros	None configured	Chapter 10, "Configuring Command Macros"
VLANs		
• Default VLAN	VLAN 1	Chapter 11, "Configuring VLANs"
• VLAN interface mode	Access	
• VLAN type	UNI isolated	
• Private VLANs	None configured	Chapter 12, "Configuring Private VLANs"
Dynamic ARP inspection (requires metro IP access or metro access image)	Disabled on all VLANs	Chapter 19, "Configuring Dynamic ARP Inspection"

Table 1-1 Default Settings After Initial Switch Configuration (continued)

Feature	Default Setting	More information in...
Tunneling		
• 802.1Q tunneling (requires metro IP access or metro access image)	Disabled	Chapter 13, "Configuring IEEE 802.1Q and Layer 2 Protocol Tunneling"
• Layer 2 protocol tunneling (requires metro IP access or metro access image)	Disabled	
Spanning Tree Protocol		
• STP	Rapid PVST+ enabled on NNIs in VLAN 1	Chapter 14, "Configuring STP"
• MSTP	Disabled (only supported on NNIs)	Chapter 15, "Configuring MSTP"
• Optional spanning-tree features	Disabled (only supported on NNIs)	Chapter 16, "Configuring Optional Spanning-Tree Features"
Flex Links (requires metro IP access or metro access image)	Not configured	Chapter 17, "Configuring Flex Links"
DHCP snooping	Disabled	Chapter 18, "Configuring DHCP Features and IP Source Guard"
IP source guard (requires metro IP access or metro access image)	Disabled	Chapter 18, "Configuring DHCP Features and IP Source Guard"
IGMP snooping		
• IGMP snooping	Enabled	Chapter 20, "Configuring IGMP Snooping and MVR"
• IGMP filters	None applied	
• IGMP querier	Disabled	
• MVR	Disabled	
IGMP throttling	Deny	Chapter 20, "Configuring IGMP Snooping and MVR"
Port-based Traffic Control		
• Broadcast, multicast, and unicast storm control	Disabled	Chapter 21, "Configuring Port-Based Traffic Control"
• Protected ports	None defined	
• Unicast and multicast traffic flooding	Not blocked	
• Secure ports	None configured	
CDP	Enabled (supported only on NNIs)	Chapter 22, "Configuring CDP"
UDLD	Disabled	Chapter 23, "Configuring UDLD"
SPAN and RSPAN	Disabled	Chapter 24, "Configuring SPAN and RSPAN"
RMON	Disabled	Chapter 25, "Configuring RMON"
Syslog messages	Enabled; displayed on the console	Chapter 26, "Configuring System Message Logging"

Table 1-1 Default Settings After Initial Switch Configuration (continued)

Feature	Default Setting	More information in...
SNMP	Enabled; Version 1	Chapter 27, "Configuring SNMP"
ACLs	None configured	Chapter 28, "Configuring Network Security with ACLs"
QoS	Not configured	Chapter 30, "Configuring QoS"
EtherChannels	None configured	Chapter 31, "Configuring EtherChannels"
IP unicast routing		
• IP routing and routing protocols (requires metro IP access or metro access image))	Disabled	Chapter 32, "Configuring IP Unicast Routing"
• Multi-VRF-CE (requires metro IP access or metro access image)	Disabled	
HSRP groups (requires metro IP access image)	None configured	Chapter 33, "Configuring HSRP"
IP multicast routing (requires metro IP access image)	Disabled on all interfaces	Chapter 34, "Configuring IP Multicast Routing"
MSDP (requires metro IP access image)	Disabled	Chapter 35, "Configuring MSDP"

Network Configuration Examples

This section provides network configuration concepts and includes examples of using the switch to create dedicated network segments and interconnecting the segments through Fast Ethernet and Gigabit Ethernet connections.

- "Multidwelling or Ethernet-to-the-Subscriber Network" section on page 1-11
- "Layer 2 VPN Application" section on page 1-13
- "Multi-VRF CE Application" section on page 1-14

Multidwelling or Ethernet-to-the-Subscriber Network

Metro Ethernet provides the access technology for service providers deploying voice, video, and Internet access services to metropolitan areas. The Metro Ethernet user-facing provider edge (UPE) switches provide economical bandwidth and the security and the QoS needed for these services.

Figure 1-1 shows a Gigabit Ethernet ring for a residential location, serving multitenant units by using Cisco ME 3400 Ethernet Access switches connected through 1000BASE-X SFP module ports. Cisco ME switches used as residential switches provide customers with high-speed connections to the service provider point-of presence (POP).

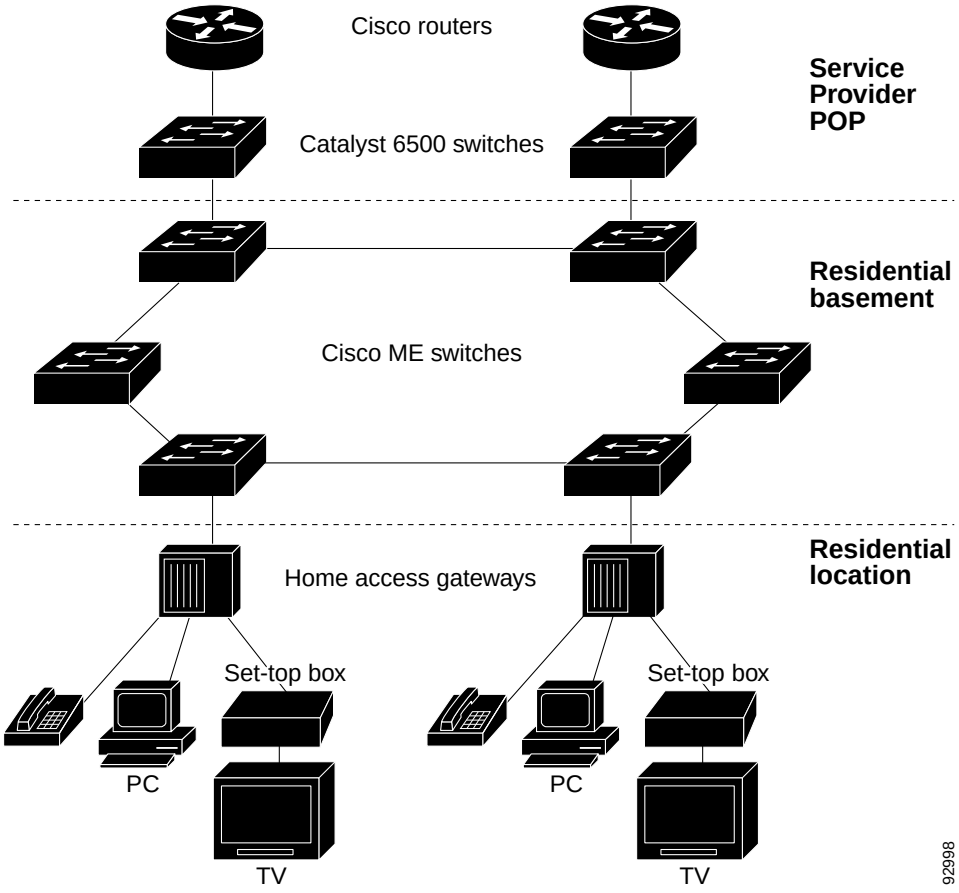
Home access gateways are connected to the ME switches through UNIs configured as 802.1Q trunks. Because the default behavior on UNIs allows no local switching between UNI ports, the subscribers are protected from each other. UNIs also do not process control protocols from customers, so

denial-of-service attacks are avoided. The Cisco ME switch also provides mechanisms such as port security and IP Source Guard to protect against MAC or IP spoofing. By using advanced access control lists, the service providers have granular control of the types of traffic to enter the network.

To provide differential QoS treatment for different types of traffic, the Cisco ME switch can identify, police, mark, and schedule traffic types based on Layer 2 to Layer 4 information. The Cisco modular QoS command-line interface (CLI), or MQC, on Cisco ME switches provides an efficient method of QoS configuration. You can configure a policer on ingress UNIs to ensure that a customer can send only the amount of bandwidth paid for. On egress NNIs, you can use four different queues to provide different levels of priority for different types of traffic. One queue can be assigned as a low-latency queue to provide expedited service for latency sensitive traffic such as voice. You can also configure a rate-limiter on the low-latency queues to prevent other queues from being deprived due to misconfiguration.

When an end station in one VLAN needs to communicate with an end station in another VLAN, a router or switch routes the traffic to the appropriate destination VLAN, providing inter-VLAN routing. VLAN access control lists (VLAN maps) provide intra-VLAN security and prevent unauthorized users from accessing critical pieces of the network. The routers also provide firewall services, Network Address Translation (NAT) services, voice-over-IP (VoIP) gateway services, and WAN and Internet access.

Figure 1-1 Cisco ME Switches in a Multidwelling Configuration



Layer 2 VPN Application

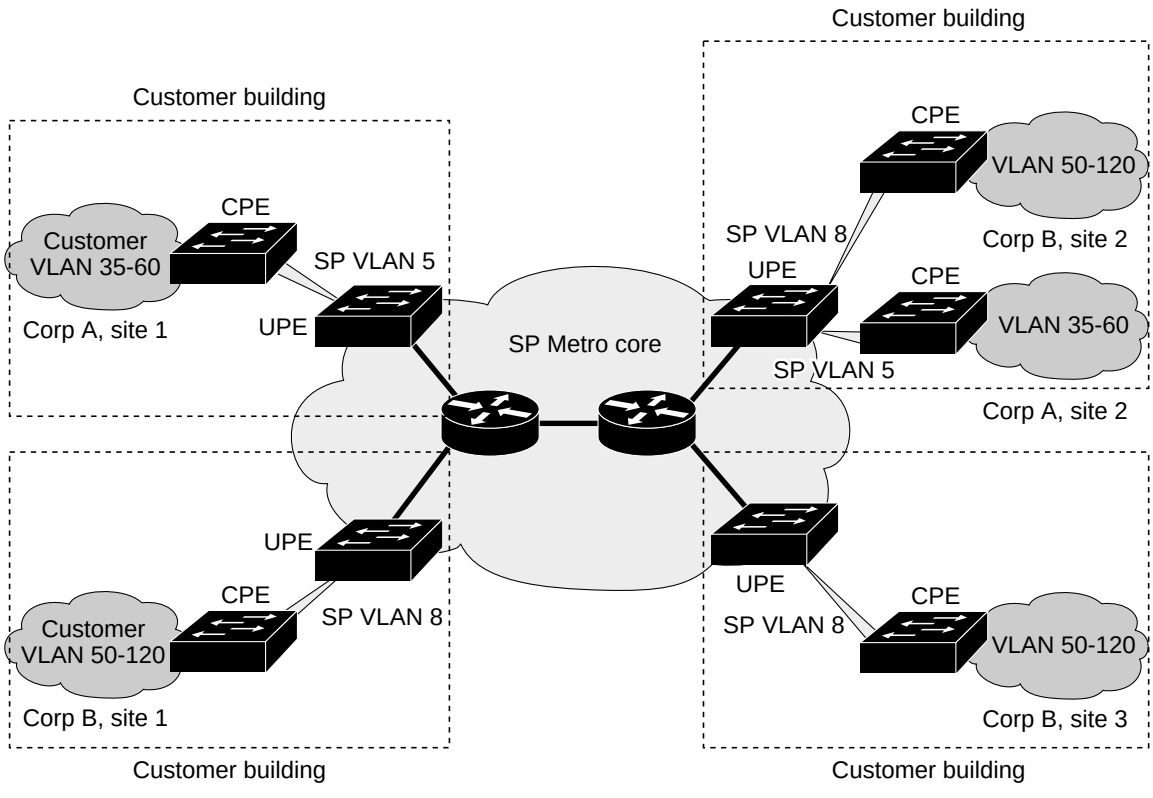
Enterprise customers need not only high bandwidth, but also the ability to extend their private network across the service provider's shared infrastructure. With Ethernet in the WAN network, service providers can meet the bandwidth requirements of enterprise customers and use VPN features to extend customers' networks.

Enterprise customers can use Layer 2 VPN to transparently move any type of traffic across a service-provider network, and create virtual pipes across the service provider infrastructure. In contrast to Layer 3 VPN service, Layer 2 VPN lowers operational expenses by minimizing enterprise user-facing provider edge (UPE) switch configuration and management. You can use Cisco ME 3400 switches to form Layer 2 VPNs so that customers at different locations can exchange information through a service-provider network without requiring dedicated connections.

In Figure 1-2, Cisco ME 3400 switches are used as UPEs in customer sites connected to customer-premises equipment (CPE) switches. The switches can tag customer traffic with the service-provider VLAN ID on top of the customer's IEEE 802.1Q tag. By supporting double tags, the Cisco ME 3400 switch provides a virtual tunnel for each customer and prevents VLAN ID overlaps between customers. In addition to data-plane separation, the Cisco ME 3400 switch can also tunnel the customer's control protocols. With Layer 2 protocol tunneling, the switch can encapsulate each customer's control-plane traffic and send it transparently across the service-provider network.

See Chapter 13, "Configuring IEEE 802.1Q and Layer 2 Protocol Tunneling," for more information on configuring these features.

Figure 1-2 Layer 2 VPN Configuration



UPE = Cisco ME 3400 switch

Multi-VRF CE Application

A VPN is a collection of sites sharing a common routing table. A customer site is connected to the service-provider network by one or more interfaces, and the service provider associates each interface with a VPN routing table, called a VPN routing/forwarding (VRF) table. Multiple VPN routing/forwarding (multi-VRF) instances in customer edge (CE) devices (multi-VRF CE) allows a service provider to support two or more VPNs with overlapping IP addresses.

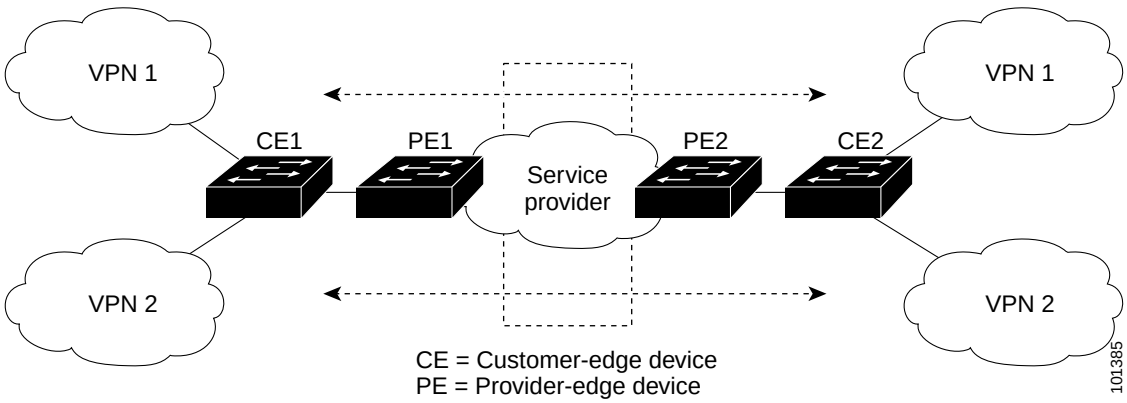
Multi-VRF CE includes these devices:

- Customer edge (CE) devices provide customers access to the service-provider network over a data link to one or more provider edge routers. The CE device advertises the site's local routes to the router and learns the remote VPN routes from the router. The Cisco ME 3400 switch can be a CE device.
- Provider edge (PE) routers exchange routing information with CE devices by using static routing or a routing protocol such as BGP, RIPv2, OSPF, or EIGRP. The PE is only required to maintain VPN routes for directly attached VPNs. It does not need to maintain all of the service-provider VPN routes. Each PE router maintains a VRF for each of its directly connected sites.
- Provider routers or core routers are any routers in the service provider network that do not attach to CE devices.

With multi-VRF CE, multiple customers can share one CE, and only one physical link is used between the CE and the PE. The shared CE maintains separate VRF tables for each customer and switches or routes packets for each customer based on its own routing table. Multi-VRF CE extends limited PE functionality to a CE device, giving it the ability to maintain separate VRF tables to extend the privacy and security of a VPN to the branch office.

Figure 1-3 shows a configuration using Cisco ME 3400 switches as multiple virtual CEs. This scenario is suited for customers who have low bandwidth requirements for their VPN service, for example, small companies. In this case, multi-VRF CE support is required in the Cisco ME switches. Because multi-VRF CE is a Layer 3 feature, each interface in a VRF must be a Layer 3 interface.

Figure 1-3 Multiple Virtual CEs



See the "Configuring Multi-VRF CE" section on page 32-59 for more information about Multi-VRF-CE.

Where to Go Next

Before configuring the switch, review these sections for startup information:

- Chapter 2, “Using the Command-Line Interface”
- Chapter 3, “Assigning the Switch IP Address and Default Gateway”
- Chapter 4, “Configuring Cisco IOS CNS Agents”



Using the Command-Line Interface

This chapter describes the Cisco IOS command-line interface (CLI) and how to use it to configure your Cisco ME 3400 Ethernet Access switch. It contains these sections:

- Understanding Command Modes, page 2-1
- Understanding the Help System, page 2-3
- Understanding Abbreviated Commands, page 2-3
- Understanding no and default Forms of Commands, page 2-4
- Understanding CLI Error Messages, page 2-4
- Using Command History, page 2-4
- Using Editing Features, page 2-6
- Searching and Filtering Output of show and more Commands, page 2-8
- Accessing the CLI, page 2-9

Understanding Command Modes

The Cisco IOS user interface is divided into many different modes. The commands available to you depend on which mode you are currently in. Enter a question mark (?) at the system prompt to obtain a list of commands available for each command mode.

When you start a session on the switch, you begin in user mode, often called user EXEC mode. Only a limited subset of the commands are available in user EXEC mode. For example, most of the user EXEC commands are one-time commands, such as **show** commands, which show the current configuration status, and **clear** commands, which clear counters or interfaces. The user EXEC commands are not saved when the switch reboots.

To have access to all commands, you must enter privileged EXEC mode. Normally, you must enter a password to enter privileged EXEC mode. From this mode, you can enter any privileged EXEC command or enter global configuration mode.

Using the configuration modes (global, interface, and line), you can make changes to the running configuration. If you save the configuration, these commands are stored and used when the switch reboots. To access the various configuration modes, you must start at global configuration mode. From global configuration mode, you can enter interface configuration mode and line configuration mode.

Table 2-1 describes the main command modes, how to access each one, the prompt you see in that mode, and how to exit the mode. The examples in the table use the hostname *Switch*.

Table 2-1 Command Mode Summary

Mode	Access Method	Prompt	Exit Method	About This Mode
User EXEC	Begin a session with your switch.	`Switch>`	Enter **logout** or **quit**.	Use this mode to • Change terminal settings. • Perform basic tests. • Display system information.
Privileged EXEC	While in user EXEC mode, enter the **enable** command.	`Switch#`	Enter **disable** to exit.	Use this mode to verify commands that you have entered. Use a password to protect access to this mode.
Global configuration	While in privileged EXEC mode, enter the **configure** command.	`Switch(config)#`	To exit to privileged EXEC mode, enter **exit** or **end**, or press **Ctrl-Z**.	Use this mode to configure parameters that apply to the entire switch.
VLAN configuration	While in global configuration mode, enter the **vlan** *vlan-id* command.	`Switch(config-vlan)#`	To exit to global configuration mode, enter the **exit** command. To return to privileged EXEC mode, press **Ctrl-Z** or enter **end**.	Use this mode to configure VLAN parameters.
Interface configuration	While in global configuration mode, enter the **interface** command (with a specific interface).	`Switch(config-if)#`	To exit to global configuration mode, enter **exit**. To return to privileged EXEC mode, press **Ctrl-Z** or enter **end**.	Use this mode to configure parameters for the Ethernet ports. For information about defining interfaces, see the "Using Interface Configuration Mode" section on page 9-7. To configure multiple interfaces with the same parameters, see the "Configuring a Range of Interfaces" section on page 9-8.
Line configuration	While in global configuration mode, specify a line with the **line vty** or **line console** command.	`Switch(config-line)#`	To exit to global configuration mode, enter **exit**. To return to privileged EXEC mode, press **Ctrl-Z** or enter **end**.	Use this mode to configure parameters for the terminal line.

For more detailed information on the command modes, see the command reference guide for this release.

Understanding the Help System

You can enter a question mark (?) at the system prompt to display a list of commands available for each command mode. You can also obtain a list of associated keywords and arguments for any command, as shown in Table 2-2.

Table 2-2 Help Summary

Command	Purpose
help	Obtain a brief description of the help system in any command mode.
*abbreviated-command-entry***?**	Obtain a list of commands that begin with a particular character string. For example: `Switch# di?` `dir disable disconnect`
abbreviated-command-entry<**Tab**>	Complete a partial command name. For example: `Switch# sh conf<tab>` `Switch# show configuration`
?	List all commands available for a particular command mode. For example: `Switch> ?`
command **?**	List the associated keywords for a command. For example: `Switch> show ?`
command keyword **?**	List the associated arguments for a keyword. For example: `Switch(config)# cdp holdtime ?` `  <10-255> Length of time (in sec) that receiver must keep this packet`

Understanding Abbreviated Commands

You need to enter only enough characters for the switch to recognize the command as unique.

This example shows how to enter the **show configuration** privileged EXEC command in an abbreviated form:

```
Switch# show conf
```

Understanding no and default Forms of Commands

Almost every configuration command also has a **no** form. In general, use the **no** form to disable a feature or function or reverse the action of a command. For example, the **no shutdown** interface configuration command reverses the shutdown of an interface. Use the command without the keyword **no** to re-enable a disabled feature or to enable a feature that is disabled by default.

Configuration commands can also have a **default** form. The **default** form of a command returns the command setting to its default. Most commands are disabled by default, so the **default** form is the same as the **no** form. However, some commands are enabled by default and have variables set to certain default values. In these cases, the **default** command enables the command and sets variables to their default values.

Understanding CLI Error Messages

Table 2-3 lists some error messages that you might encounter while using the CLI to configure your switch.

Table 2-3 Common CLI Error Messages

Error Message	Meaning	How to Get Help
`% Ambiguous command: "show con"`	You did not enter enough characters for your switch to recognize the command.	Re-enter the command followed by a question mark (?) with a space between the command and the question mark. The possible keywords that you can enter with the command appear.
`% Incomplete command.`	You did not enter all the keywords or values required by this command.	Re-enter the command followed by a question mark (?) with a space between the command and the question mark. The possible keywords that you can enter with the command appear.
`% Invalid input detected at '^' marker.`	You entered the command incorrectly. The caret (^) marks the point of the error.	Enter a question mark (?) to display all the commands that are available in this command mode. The possible keywords that you can enter with the command appear.

Using Command History

The software provides a history or record of commands that you have entered. The command history feature is particularly useful for recalling long or complex commands or entries, including access lists. You can customize this feature to suit your needs as described in these sections:

- Changing the Command History Buffer Size, page 2-5 (optional)
- Recalling Commands, page 2-5 (optional)
- Disabling the Command History Feature, page 2-5 (optional)

Changing the Command History Buffer Size

By default, the switch records ten command lines in its history buffer. You can alter this number for a current terminal session or for all sessions on a particular line. These procedures are optional.

Beginning in privileged EXEC mode, enter this command to change the number of command lines that the switch records during the current terminal session:

```
Switch# terminal history [size number-of-lines]
```

The range is from 0 to 256.

Beginning in line configuration mode, enter this command to configure the number of command lines the switch records for all sessions on a particular line:

```
Switch(config-line)# history [size number-of-lines]
```

The range is from 0 to 256.

Recalling Commands

To recall commands from the history buffer, perform one of the actions listed in Table 2-4. These actions are optional.

Table 2-4 Recalling Commands

Action[1]	Result
Press **Ctrl-P** or the up arrow key.	Recall commands in the history buffer, beginning with the most recent command. Repeat the key sequence to recall successively older commands.
Press **Ctrl-N** or the down arrow key.	Return to more recent commands in the history buffer after recalling commands with **Ctrl-P** or the up arrow key. Repeat the key sequence to recall successively more recent commands.
show history	While in privileged EXEC mode, list the last several commands that you just entered. The number of commands that appear is controlled by the setting of the **terminal history** global configuration command and the **history** line configuration command.

1. The arrow keys function only on ANSI-compatible terminals such as VT100s.

Disabling the Command History Feature

The command history feature is automatically enabled. You can disable it for the current terminal session or for the command line. These procedures are optional.

To disable the feature during the current terminal session, enter the **terminal no history** privileged EXEC command.

To disable command history for the line, enter the **no history** line configuration command.

Using Editing Features

This section describes the editing features that can help you manipulate the command line. It contains these sections:

- Enabling and Disabling Editing Features, page 2-6 (optional)
- Editing Commands through Keystrokes, page 2-6 (optional)
- Editing Command Lines that Wrap, page 2-8 (optional)

Enabling and Disabling Editing Features

Although enhanced editing mode is automatically enabled, you can disable it, re-enable it, or configure a specific line to have enhanced editing. These procedures are optional.

To globally disable enhanced editing mode, enter this command in line configuration mode:

```
Switch (config-line)# no editing
```

To re-enable the enhanced editing mode for the current terminal session, enter this command in privileged EXEC mode:

```
Switch# terminal editing
```

To reconfigure a specific line to have enhanced editing mode, enter this command in line configuration mode:

```
Switch(config-line)# editing
```

Editing Commands through Keystrokes

Table 2-5 shows the keystrokes that you need to edit command lines. These keystrokes are optional.

Table 2-5 Editing Commands through Keystrokes

Capability	Keystroke[1]	Purpose
Move around the command line to make changes or corrections.	Press **Ctrl-B**, or press the left arrow key.	Move the cursor back one character.
	Press **Ctrl-F**, or press the right arrow key.	Move the cursor forward one character.
	Press **Ctrl-A**.	Move the cursor to the beginning of the command line.
	Press **Ctrl-E**.	Move the cursor to the end of the command line.
	Press **Esc B**.	Move the cursor back one word.
	Press **Esc F**.	Move the cursor forward one word.
	Press **Ctrl-T**.	Transpose the character to the left of the cursor with the character located at the cursor.
Recall commands from the buffer and paste them in the command line. The switch provides a buffer with the last ten items that you deleted.	Press **Ctrl-Y**.	Recall the most recent entry in the buffer.

Table 2-5 Editing Commands through Keystrokes (continued)

Capability	Keystroke[1]	Purpose
	Press **Esc Y**.	Recall the next buffer entry. The buffer contains only the last 10 items that you have deleted or cut. If you press **Esc Y** more than ten times, you cycle to the first buffer entry.
Delete entries if you make a mistake or change your mind.	Press the **Delete** or **Backspace** key.	Erase the character to the left of the cursor.
	Press **Ctrl-D**.	Delete the character at the cursor.
	Press **Ctrl-K**.	Delete all characters from the cursor to the end of the command line.
	Press **Ctrl-U** or **Ctrl-X**.	Delete all characters from the cursor to the beginning of the command line.
	Press **Ctrl-W**.	Delete the word to the left of the cursor.
	Press **Esc D**.	Delete from the cursor to the end of the word.
Capitalize or lowercase words or capitalize a set of letters.	Press **Esc C**.	Capitalize at the cursor.
	Press **Esc L**.	Change the word at the cursor to lowercase.
	Press **Esc U**.	Capitalize letters from the cursor to the end of the word.
Designate a particular keystroke as an executable command, perhaps as a shortcut.	Press **Ctrl-V** or **Esc Q**.	
Scroll down a line or screen on displays that are longer than the terminal screen can display. **Note** The More prompt is used for any output that has more lines than can be displayed on the terminal screen, including **show** command output. You can use the **Return** and **Space** bar keystrokes whenever you see the More prompt.	Press the **Return** key.	Scroll down one line.
	Press the **Space** bar.	Scroll down one screen.
Redisplay the current command line if the switch suddenly sends a message to your screen.	Press **Ctrl-L** or **Ctrl-R**.	Redisplay the current command line.

1. The arrow keys function only on ANSI-compatible terminals such as VT100s.

Editing Command Lines that Wrap

You can use a wraparound feature for commands that extend beyond a single line on the screen. When the cursor reaches the right margin, the command line shifts ten spaces to the left. You cannot see the first ten characters of the line, but you can scroll back and check the syntax at the beginning of the command. The keystroke actions are optional.

To scroll back to the beginning of the command entry, press **Ctrl-B** or the left arrow key repeatedly. You can also press **Ctrl-A** to immediately move to the beginning of the line.

Note The arrow keys function only on ANSI-compatible terminals such as VT100s.

In this example, the **access-list** global configuration command entry extends beyond one line. When the cursor first reaches the end of the line, the line is shifted ten spaces to the left and redisplayed. The dollar sign ($) shows that the line has been scrolled to the left. Each time the cursor reaches the end of the line, the line is again shifted ten spaces to the left.

```
Switch(config)# access-list 101 permit tcp 131.108.2.5 255.255.255.0 131.108.1
Switch(config)# $ 101 permit tcp 131.108.2.5 255.255.255.0 131.108.1.20 255.25
Switch(config)# $t tcp 131.108.2.5 255.255.255.0 131.108.1.20 255.255.255.0 eq
Switch(config)# $108.2.5 255.255.255.0 131.108.1.20 255.255.255.0 eq 45
```

After you complete the entry, press **Ctrl-A** to check the complete syntax before pressing the **Return** key to execute the command. The dollar sign ($) appears at the end of the line to show that the line has been scrolled to the right:

```
Switch(config)# access-list 101 permit tcp 131.108.2.5 255.255.255.0 131.108.1$
```

The software assumes you have a terminal screen that is 80 columns wide. If you have a width other than that, use the **terminal width** privileged EXEC command to set the width of your terminal.

Use line wrapping with the command history feature to recall and modify previous complex command entries. For information about recalling previous command entries, see the "Editing Commands through Keystrokes" section on page 2-6.

Searching and Filtering Output of show and more Commands

You can search and filter the output for **show** and **more** commands. This is useful when you need to sort through large amounts of output or if you want to exclude output that you do not need to see. Using these commands is optional.

To use this functionality, enter a **show** or **more** command followed by the *pipe* character (|), one of the keywords **begin**, **include**, or **exclude**, and an expression that you want to search for or filter out:

command **|** {**begin** | **include** | **exclude**} *regular-expression*

Expressions are case sensitive. For example, if you enter **| exclude output**, the lines that contain *output* are not displayed, but the lines that contain *Output* appear.

This example shows how to include in the output display only lines where the expression *protocol* appears:

```
Switch# show interfaces | include protocol
Vlan1 is up, line protocol is up
Vlan10 is up, line protocol is down
GigabitEthernet0/1 is up, line protocol is down
GigabitEthernet0/2 is up, line protocol is up
```

Accessing the CLI

You can access the CLI through a console connection, through Telnet, or by using the browser.

Accessing the CLI through a Console Connection or through Telnet

Before you can access the CLI, you must connect a terminal or PC to the switch console port and power on the switch as described in the hardware installation guide that shipped with your switch. Then, to understand the boot process and the options available for assigning IP information, see Chapter 3, "Assigning the Switch IP Address and Default Gateway."

If your switch is already configured, you can access the CLI through a local console connection or through a remote Telnet session, but your switch must first be configured for this type of access. For more information, see the "Setting a Telnet Password for a Terminal Line" section on page 7-6.

You can use one of these methods to establish a connection with the switch:

- Connect the switch console port to a management station or dial-up modem. For information about connecting to the console port, see the switch hardware installation guide.
- Use any Telnet TCP/IP or encrypted Secure Shell (SSH) package from a remote management station. The switch must have network connectivity with the Telnet or SSH client, and the switch must have an enable secret password configured.

 For information about configuring the switch for Telnet access, see the "Setting a Telnet Password for a Terminal Line" section on page 7-6. The switch supports up to 16 simultaneous Telnet sessions. Changes made by one Telnet user are reflected in all other Telnet sessions.

 For information about configuring the switch for SSH, see the "Configuring the Switch for Secure Shell" section on page 7-37. The switch supports up to five simultaneous secure SSH sessions.

After you connect through the console port, through a Telnet session or through an SSH session, the user EXEC prompt appears on the management station.



Assigning the Switch IP Address and Default Gateway

This chapter describes how to create the initial switch configuration (for example, assigning the switch IP address and default gateway information) for the Cisco Metro Ethernet (ME) 3400 Ethernet Access switch by using a variety of automatic and manual methods. It also describes how to modify the switch startup configuration.

Note For complete syntax and usage information for the commands used in this chapter, see the command reference for this release and to the *Cisco IOS IP Command Reference, Volume 1 of 3: Addressing and Services, Release 12.2*.

This chapter consists of these sections:

- Understanding the Boot Process, page 3-1
- Assigning Switch Information, page 3-2
- Checking and Saving the Running Configuration, page 3-10
- Modifying the Startup Configuration, page 3-12
- Scheduling a Reload of the Software Image, page 3-16

Note Information in this chapter about configuring IP addresses and DHCP is specific to IP Version 4 (IPv4).

Understanding the Boot Process

To start your switch, you need to follow the procedures in the hardware installation guide about installing and powering on the switch, and setting up the initial configuration (IP address, subnet mask, default gateway, secret and Telnet passwords, and so forth) of the switch.

The normal boot process involves the operation of the boot loader software, which performs these functions:

- Performs low-level CPU initialization. It initializes the CPU registers, which control where physical memory is mapped, its quantity, its speed, and so forth.
- Performs power-on self-test (POST) for the CPU subsystem. It tests the CPU DRAM and the portion of the flash device that makes up the flash file system.

- Initializes the flash file system on the system board.
- Loads a default operating system software image into memory and boots the switch.

The boot loader provides access to the flash file system before the operating system is loaded. Normally, the boot loader is used only to load, uncompress, and launch the operating system. After the boot loader gives the operating system control of the CPU, the boot loader is not active until the next system reset or power-on.

The boot loader also provides trap-door access into the system if the operating system has problems serious enough that it cannot be used. The trap-door mechanism provides enough access to the system so that if it is necessary, you can format the flash file system, reinstall the operating system software image by using the XMODEM Protocol, recover from a lost or forgotten password, and finally restart the operating system. For more information, see the "Recovering from Corrupted Software By Using the Xmodem Protocol" section on page 36-2 and the "Recovering from a Lost or Forgotten Password" section on page 36-3.

Note You can disable password recovery. For more information, see the "Disabling Password Recovery" section on page 7-5.

Before you can assign switch information, make sure you have connected a PC or terminal to the console port, and configured the PC or terminal-emulation software baud rate and character format to match these of the switch console port:

- Baud rate default is 9600.
- Data bits default is 8.

 Note If the data bits option is set to 8, set the parity option to none.

- Stop bits default is 1.
- Parity settings default is none.

Assigning Switch Information

You can assign IP information through the switch setup program, through a DHCP server, or manually.

Use the switch setup program if you want to be prompted for specific IP information. With this program, you can also configure a hostname and an enable secret password. It gives you the option of assigning a Telnet password (to provide security during remote management). For more information about the setup program, see the "Configuring the Switch with the CLI-Based Setup Program" appendix in the hardware installation guide.

Use a DHCP server for centralized control and automatic assignment of IP information after the server is configured.

Note If you are using DHCP, do not respond to any of the questions in the setup program until the switch receives the dynamically assigned IP address and reads the configuration file.

If you are an experienced user familiar with the switch configuration steps, manually configure the switch. Otherwise, use the setup program described previously.

These sections contain this configuration information:

- Default Switch Information, page 3-3
- Understanding DHCP-Based Autoconfiguration, page 3-3
- Manually Assigning IP Information, page 3-9

Default Switch Information

Table 3-1 shows the default switch information.

Table 3-1 Default Switch Information

Feature	Default Setting
IP address and subnet mask	No IP address or subnet mask are defined.
Default gateway	No default gateway is defined.
Enable secret password	No password is defined.
Hostname	The factory-assigned default hostname is *Switch.*
Telnet password	No password is defined.

Understanding DHCP-Based Autoconfiguration

DHCP provides configuration information to Internet hosts and internetworking devices. This protocol consists of two components: one for delivering configuration parameters from a DHCP server to a device and a mechanism for allocating network addresses to devices. DHCP is built on a client-server model, in which designated DHCP servers allocate network addresses and deliver configuration parameters to dynamically configured devices. The switch can act as a DHCP client, but it cannot be a DHCP server.

During DHCP-based autoconfiguration, your switch (DHCP client) is automatically configured at startup with IP address information and a configuration file.

With DHCP-based autoconfiguration, no DHCP client-side configuration is needed on your switch. However, you need to configure the DHCP server for various lease options associated with IP addresses. If you are using DHCP to relay the configuration file location on the network, you might also need to configure a Trivial File Transfer Protocol (TFTP) server and a Domain Name System (DNS) server.

The DHCP server for your switch can be on the same LAN or on a different LAN than the switch. If the DHCP server is running on a different LAN, you should configure a DHCP relay device between your switch and the DHCP server. A relay device forwards broadcast traffic between two directly connected LANs. A router does not forward broadcast packets, but it forwards packets based on the destination IP address in the received packet.

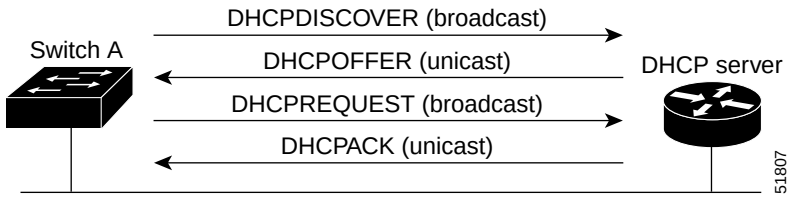
DHCP-based autoconfiguration replaces the BOOTP client functionality on your switch.

DHCP Client Request Process

When you boot your switch, the DHCP client is invoked and requests configuration information from a DHCP server when the configuration file is not present on the switch. If the configuration file is present and the configuration includes the **ip address dhcp** interface configuration command on specific routed interfaces, the DHCP client is invoked and requests the IP address information for those interfaces.

Figure 3-1 shows the sequence of messages that are exchanged between the DHCP client and the DHCP server.

Figure 3-1 DHCP Client and Server Message Exchange



The client, Switch A, broadcasts a DHCPDISCOVER message to locate a DHCP server. The DHCP server offers configuration parameters (such as an IP address, subnet mask, gateway IP address, DNS IP address, a lease for the IP address, and so forth) to the client in a DHCPOFFER unicast message.

In a DHCPREQUEST broadcast message, the client returns a formal request for the offered configuration information to the DHCP server. The formal request is broadcast so that all other DHCP servers that received the DHCPDISCOVER broadcast message from the client can reclaim the IP addresses that they offered to the client.

The DHCP server confirms that the IP address has been allocated to the client by returning a DHCPACK unicast message to the client. With this message, the client and server are bound, and the client uses configuration information received from the server. The amount of information the switch receives depends on how you configure the DHCP server. For more information, see the "Configuring the TFTP Server" section on page 3-5.

If the configuration parameters sent to the client in the DHCPOFFER unicast message are invalid (a configuration error exists), the client returns a DHCPDECLINE broadcast message to the DHCP server.

The DHCP server sends the client a DHCPNAK denial broadcast message, which means that the offered configuration parameters have not been assigned, that an error has occurred during the negotiation of the parameters, or that the client has been slow in responding to the DHCPOFFER message (the DHCP server assigned the parameters to another client).

A DHCP client might receive offers from multiple DHCP or BOOTP servers and can accept any of the offers; however, the client usually accepts the first offer it receives. The offer from the DHCP server is not a guarantee that the IP address is allocated to the client; however, the server usually reserves the address until the client has had a chance to formally request the address. If the switch accepts replies from a BOOTP server and configures itself, the switch broadcasts, instead of unicasts, TFTP requests to obtain the switch configuration file.

Configuring DHCP-Based Autoconfiguration

These sections contain this configuration information:

- DHCP Server Configuration Guidelines, page 3-5
- Configuring the TFTP Server, page 3-5
- Configuring the DNS, page 3-6
- Configuring the Relay Device, page 3-6
- Obtaining Configuration Files, page 3-7
- Example Configuration, page 3-8

If your DHCP server is a Cisco device, see the "Configuring DHCP" section of the "IP Addressing and Services" section of the *Cisco IOS IP Configuration Guide, Release 12.2* for additional information about configuring DHCP.

DHCP Server Configuration Guidelines

Follow these guidelines if you are configuring another device as a DHCP server:

You should configure the DHCP server with reserved leases that are bound to each switch by the switch hardware address.

If you want the switch to receive IP address information, you must configure the DHCP server with these lease options:

- IP address of the client (required)
- Subnet mask of the client (required)
- DNS server IP address (optional)
- Router IP address (default gateway address to be used by the switch) (required)

If you want the switch to receive the configuration file from a TFTP server, you must configure the DHCP server with these lease options:

- TFTP server name (required)
- Boot filename (the name of the configuration file that the client needs) (recommended)
- Hostname (optional)

Depending on the settings of the DHCP server, the switch can receive IP address information, the configuration file, or both.

If you do not configure the DHCP server with the lease options described previously, it replies to client requests with only those parameters that are configured. If the IP address and the subnet mask are not in the reply, the switch is not configured. If the router IP address or the TFTP server name are not found, the switch might send broadcast, instead of unicast, TFTP requests. Unavailability of other lease options does not affect autoconfiguration.

Configuring the TFTP Server

Based on the DHCP server configuration, the switch attempts to download one or more configuration files from the TFTP server. If you configured the DHCP server to respond to the switch with all the options required for IP connectivity to the TFTP server, and if you configured the DHCP server with a TFTP server name, address, and configuration filename, the switch attempts to download the specified configuration file from the specified TFTP server.

If you did not specify the configuration filename, the TFTP server, or if the configuration file could not be downloaded, the switch attempts to download a configuration file by using various combinations of filenames and TFTP server addresses. The files include the specified configuration filename (if any) and these files: network-config, cisconet.cfg, *hostname*.config, or *hostname*.cfg, where *hostname* is the switch's current hostname. The TFTP server addresses used include the specified TFTP server address (if any) and the broadcast address (255.255.255.255).

For the switch to successfully download a configuration file, the TFTP server must contain one or more configuration files in its base directory. The files can include these files:

- The configuration file named in the DHCP reply (the actual switch configuration file).
- The network-confg or the cisconet.cfg file (known as the default configuration files).
- The router-confg or the ciscortr.cfg file (These files contain commands common to all switches. Normally, if the DHCP and TFTP servers are properly configured, these files are not accessed.)

If you specify the TFTP server name in the DHCP server-lease database, you must also configure the TFTP server name-to-IP-address mapping in the DNS-server database.

If the TFTP server to be used is on a different LAN from the switch, or if it is to be accessed by the switch through the broadcast address (which occurs if the DHCP server response does not contain all the required information described previously), a relay must be configured to forward the TFTP packets to the TFTP server. For more information, see the "Configuring the Relay Device" section on page 3-6. The preferred solution is to configure the DHCP server with all the required information.

Configuring the DNS

The DHCP server uses the DNS server to resolve the TFTP server name to an IP address. You must configure the TFTP server name-to-IP address map on the DNS server. The TFTP server contains the configuration files for the switch.

You can configure the IP addresses of the DNS servers in the lease database of the DHCP server from where the DHCP replies will retrieve them. You can enter up to two DNS server IP addresses in the lease database.

The DNS server can be on the same or on a different LAN as the switch. If it is on a different LAN, the switch must be able to access it through a router.

Configuring the Relay Device

You must configure a relay device, also referred to as a *relay agent*, when a switch sends broadcast packets that require a response from a host on a different LAN. Examples of broadcast packets that the switch might send are DHCP, DNS, and in some cases, TFTP packets. You must configure this relay device to forward received broadcast packets on an interface to the destination host.

If the relay device is a Cisco router, enable IP routing (**ip routing** global configuration command), and configure helper addresses by using the **ip helper-address** interface configuration command.

For example, in Figure 3-2, configure the router interfaces as follows:

On interface 10.0.0.2:

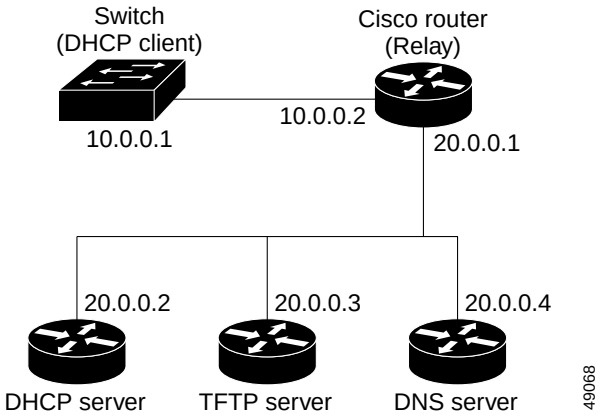
```
router(config-if)# ip helper-address 20.0.0.2
router(config-if)# ip helper-address 20.0.0.3
router(config-if)# ip helper-address 20.0.0.4
```

On interface 20.0.0.1

```
router(config-if)# ip helper-address 10.0.0.1
```

Note If the switch is acting as the relay device, configure the interface as a routed port. For more information, see the "Routed Ports" section on page 9-4 and the "Configuring Layer 3 Interfaces" section on page 9-20.

Figure 3-2 Relay Device Used in Autoconfiguration



Obtaining Configuration Files

Depending on the availability of the IP address and the configuration filename in the DHCP reserved lease, the switch obtains its configuration information in these ways:

- The IP address and the configuration filename is reserved for the switch and provided in the DHCP reply (one-file read method).

 The switch receives its IP address, subnet mask, TFTP server address, and the configuration filename from the DHCP server. The switch sends a unicast message to the TFTP server to retrieve the named configuration file from the base directory of the server and upon receipt, it completes its boot-up process.

- The IP address and the configuration filename is reserved for the switch, but the TFTP server address is not provided in the DHCP reply (one-file read method).

 The switch receives its IP address, subnet mask, and the configuration filename from the DHCP server. The switch sends a broadcast message to a TFTP server to retrieve the named configuration file from the base directory of the server, and upon receipt, it completes its boot-up process.

- Only the IP address is reserved for the switch and provided in the DHCP reply. The configuration filename is not provided (two-file read method).

 The switch receives its IP address, subnet mask, and the TFTP server address from the DHCP server. The switch sends a unicast message to the TFTP server to retrieve the network-confg or cisconet.cfg default configuration file. (If the network-confg file cannot be read, the switch reads the cisconet.cfg file.)

 The default configuration file contains the hostnames-to-IP-address mapping for the switch. The switch fills its host table with the information in the file and obtains its hostname. If the hostname is not found in the file, the switch uses the hostname in the DHCP reply. If the hostname is not specified in the DHCP reply, the switch uses the default *Switch* as its hostname.

After obtaining its hostname from the default configuration file or the DHCP reply, the switch reads the configuration file that has the same name as its hostname (*hostname*-confg or *hostname*.cfg, depending on whether network-confg or cisconet.cfg was read earlier) from the TFTP server. If the cisconet.cfg file is read, the filename of the host is truncated to eight characters.

If the switch cannot read the network-confg, cisconet.cfg, or the hostname file, it reads the router-confg file. If the switch cannot read the router-confg file, it reads the ciscortr.cfg file.

Note The switch broadcasts TFTP server requests if the TFTP server is not obtained from the DHCP replies, if all attempts to read the configuration file through unicast transmissions fail, or if the TFTP server name cannot be resolved to an IP address.

Example Configuration

Figure 3-3 shows a sample network for retrieving IP information by using DHCP-based autoconfiguration.

Figure 3-3 DHCP-Based Autoconfiguration Network Example

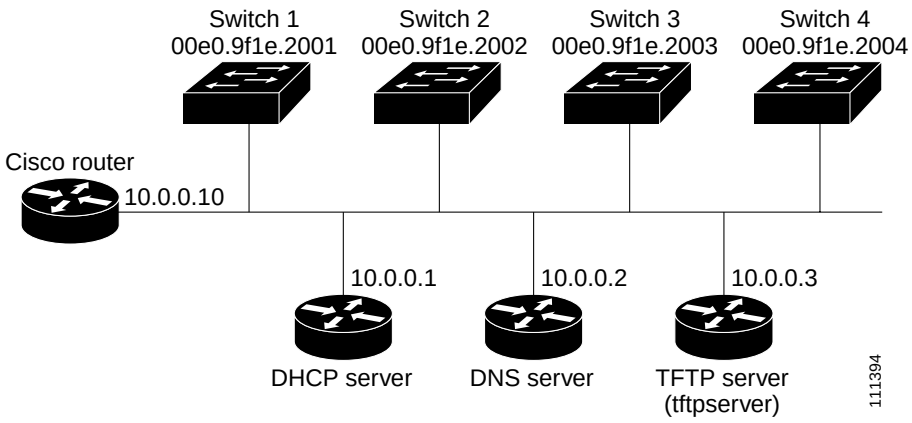


Table 3-2 shows the configuration of the reserved leases on the DHCP server.

Table 3-2 DHCP Server Configuration

	Switch A	Switch B	Switch C	Switch D
Binding key (hardware address)	00e0.9f1e.2001	00e0.9f1e.2002	00e0.9f1e.2003	00e0.9f1e.2004
IP address	10.0.0.21	10.0.0.22	10.0.0.23	10.0.0.24
Subnet mask	255.255.255.0	255.255.255.0	255.255.255.0	255.255.255.0
Router address	10.0.0.10	10.0.0.10	10.0.0.10	10.0.0.10
DNS server address	10.0.0.2	10.0.0.2	10.0.0.2	10.0.0.2
TFTP server name	tftpserver or 10.0.0.3	tftpserver or 10.0.0.3	tftpserver or 10.0.0.3	tftpserver or 10.0.0.3
Boot filename (configuration file) (optional)	switcha-confg	switchb-confg	switchc-confg	switchd-confg
Hostname (optional)	switcha	switchb	switchc	switchd

DNS Server Configuration

The DNS server maps the TFTP server name *tftpserver* to IP address 10.0.0.3.

TFTP Server Configuration (on UNIX)

The TFTP server base directory is set to /tftpserver/work/. This directory contains the network-confg file used in the two-file read method. This file contains the hostname to be assigned to the switch based on its IP address. The base directory also contains a configuration file for each switch (*switcha-confg*, *switchb-confg*, and so forth) as shown in this display:

```
prompt> cd /tftpserver/work/
prompt> ls
network-confg
switcha-confg
switchb-confg
switchc-confg
switchd-confg
prompt> cat network-confg
ip host switcha 10.0.0.21
ip host switchb 10.0.0.22
ip host switchc 10.0.0.23
ip host switchd 10.0.0.24
```

DHCP Client Configuration

No configuration file is present on Switch A through Switch D.

Configuration Explanation

In Figure 3-3, Switch A reads its configuration file as follows:

- It obtains its IP address 10.0.0.21 from the DHCP server.
- If no configuration filename is given in the DHCP server reply, Switch A reads the network-confg file from the base directory of the TFTP server.
- It adds the contents of the network-confg file to its host table.
- It reads its host table by indexing its IP address 10.0.0.21 to its hostname (switcha).
- It reads the configuration file that corresponds to its hostname; for example, it reads *switch1-confg* from the TFTP server.

Switches B through D retrieve their configuration files and IP addresses in the same way.

Manually Assigning IP Information

Beginning in privileged EXEC mode, follow these steps to manually assign IP information to a switch virtual interface (SVI). If the switch is running the metro IP access image, you can also manually assign IP information to a port if you first put the port into Layer 3 mode by using the **no switchport** command.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface vlan** *vlan-id*	Enter interface configuration mode, and enter the VLAN to which the IP information is assigned. The range is 1 to 4094; do not enter leading zeros.
Step 3	**ip address** *ip-address subnet-mask*	Enter the IP address and subnet mask.
Step 4	**exit**	Return to global configuration mode.

	Command	Purpose
Step 5	**ip default-gateway** *ip-address*	Enter the IP address of the next-hop router interface that is directly connected to the switch where a default gateway is being configured. The default gateway receives IP packets with unresolved destination IP addresses from the switch. Once the default gateway is configured, the switch has connectivity to the remote networks with which a host needs to communicate. **Note** When your switch is configured to route with IP, it does not need to have a default gateway set.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show interfaces vlan** *vlan-id*	Verify the configured IP address.
Step 8	**show ip redirects**	Verify the configured default gateway.
Step 9	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the switch IP address, use the **no ip address** interface configuration command. If you are removing the address through a Telnet session, your connection to the switch will be lost. To remove the default gateway address, use the **no ip default-gateway** global configuration command.

For information on setting the switch system name, protecting access to privileged EXEC commands, and setting time and calendar services, see Chapter 5, “Administering the Switch.”

Checking and Saving the Running Configuration

You can check the configuration settings you entered or changes you made by entering this privileged EXEC command:

```
Switch# show running-config
Building configuration...

Current configuration : 2010 bytes
!
version 12.2
no service pad
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname 3400-3
!
enable password cisco
!
no aaa new-model
ip subnet-zero
no ip domain-lookup
!
table-map test
 default copy
!
no file verify auto
!
spanning-tree mode rapid-pvst
spanning-tree extend system-id
!
vlan internal allocation policy ascending
```

```
!
vlan 2,10
!
class-map match-all test1
class-map match-all class2
class-map match-all class1
!
!
policy-map test
  class class1
   police cir percent 30
policy-map test2
  class class2
   police cir 8500 bc 1500
policy-map test3
!
!
interface FastEthernet0/1
!
interface FastEthernet0/2
 shutdown
!
interface FastEthernet0/3
 shutdown
!
interface FastEthernet0/4
 shutdown
!
interface FastEthernet0/5
 shutdown
!
interface FastEthernet0/6
 shutdown
!
interface FastEthernet0/7
 shutdown

<output truncated>

interface GigabitEthernet0/1
 port-type nni
!
interface GigabitEthernet0/2
 port-type nni
!
interface Vlan1
 no ip address
 no ip route-cache
 no ip mroute-cache
 shutdown
!
interface Vlan10
 ip address 192.168.1.76 255.255.255.0
!
ip default-gateway 192.168.1.3
no ip http server
ip classless
!
!
!
control-plane
!
!
line con 0
```

```
 session-timeout 120
 exec-timeout 120 0
 speed 115200
line vty 0 4
 password cisco
 no login
line vty 5 15
 no login
!
!
end
```

To store the configuration or changes you have made to your startup configuration in flash memory, enter this privileged EXEC command:

```
Switch# copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
```

This command saves the configuration settings that you made. If you fail to do this, your configuration will be lost the next time you reload the system. To display information stored in the NVRAM section of flash memory, use the **show startup-config** or **more startup-config** privileged EXEC command.

For more information about alternative locations from which to copy the configuration file, see Appendix B, "Working with the Cisco IOS File System, Configuration Files, and Software Images."

Modifying the Startup Configuration

These sections describe how to modify the switch startup configuration:

- Default Boot Configuration, page 3-13
- Automatically Downloading a Configuration File, page 3-13
- Booting Manually, page 3-14
- Booting a Specific Software Image, page 3-14
- Controlling Environment Variables, page 3-15

See also Appendix B, "Working with the Cisco IOS File System, Configuration Files, and Software Images," for information about switch configuration files.

Default Boot Configuration

Table 3-3 shows the default boot configuration.

Table 3-3 Default Boot Configuration

Feature	Default Setting
Operating system software image	The switch attempts to automatically boot the system using information in the BOOT environment variable. If the variable is not set, the switch attempts to load and execute the first executable image it can by performing a recursive, depth-first search throughout the flash file system. The Cisco IOS image is stored in a directory that has the same name as the image file (excluding the .bin extension). In a depth-first search of a directory, each encountered subdirectory is completely searched before continuing the search in the original directory.
Configuration file	Configured switches use the *config.text* file stored on the system board in flash memory. A new switch has no configuration file.

Automatically Downloading a Configuration File

You can automatically download a configuration file to your switch by using the DHCP-based autoconfiguration feature. For more information, see the "Understanding DHCP-Based Autoconfiguration" section on page 3-3.

Specifying the Filename to Read and Write the System Configuration

By default, the Cisco IOS software uses the file *config.text* to read and write a nonvolatile copy of the system configuration. However, you can specify a different filename, which will be loaded during the next boot cycle.

Beginning in privileged EXEC mode, follow these steps to specify a different configuration filename:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**boot config-file flash:***/file-url*	Specify the configuration file to load during the next boot cycle. For *file-url*, specify the path (directory) and the configuration filename. Filenames and directory names are case sensitive.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show boot**	Verify your entries. The **boot config-file** global configuration command changes the setting of the CONFIG_FILE environment variable.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no boot config-file** global configuration command.

Booting Manually

By default, the switch automatically boots; however, you can configure it to manually boot.

Beginning in privileged EXEC mode, follow these steps to configure the switch to manually boot during the next boot cycle:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**boot manual**	Enable the switch to manually boot during the next boot cycle.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show boot**	Verify your entries. The **boot manual** global command changes the setting of the MANUAL_BOOT environment variable. The next time you reboot the system, the switch is in boot loader mode, shown by the *switch:* prompt. To boot the system, use the **boot** *filesystem***:**/*file-url* boot loader command. • For *filesystem***:**, use **flash:** for the system board flash device. • For *file-url*, specify the path (directory) and the name of the bootable image. Filenames and directory names are case sensitive.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable manual booting, use the **no boot manual** global configuration command.

Booting a Specific Software Image

By default, the switch attempts to automatically boot the system using information in the BOOT environment variable. If this variable is not set, the switch attempts to load and execute the first executable image it can by performing a recursive, depth-first search throughout the flash file system. In a depth-first search of a directory, each encountered subdirectory is completely searched before continuing the search in the original directory. However, you can specify a specific image to boot.

Beginning in privileged EXEC mode, follow these steps to configure the switch to boot a specific image during the next boot cycle:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**boot system** *filesystem***:**/*file-url*	Configure the switch to boot a specific image in flash memory during the next boot cycle. • For *filesystem***:**, use **flash:** for the system board flash device. • For *file-url*, specify the path (directory) and the name of the bootable image. Filenames and directory names are case sensitive.

	Command	Purpose
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show boot**	Verify your entries. The **boot system** global command changes the setting of the BOOT environment variable. During the next boot cycle, the switch attempts to automatically boot the system using information in the BOOT environment variable.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no boot system** global configuration command.

Controlling Environment Variables

With a normally operating switch, you enter the boot loader mode only through a switch console connection configured for 9600 bps. Unplug and then reconnect the switch power cord. After the switch performs POST, the switch begins the autoboot process. The boot loader prompts the user for a break key character during the boot-up sequence, as shown in this example:

```
***** The system will autoboot in 5 seconds *****

Send a break key to prevent autobooting.
```

The break key character is different for each operating system.

- On a SUN work station running UNIX, Ctrl-C is the break key.
- On a PC running Windows 2000, Ctrl-Break is the break key.

Cisco TAC has tabulated break keys for most common operating systems and provided an alternative break key sequence for terminal emulators that do not support the break keys. To view this table, see:

http://www.cisco.com/warp/public/701/61.html#how-to

When you enter the break key, the boot loader *switch:* prompt appears.

The switch boot loader software provides support for nonvolatile environment variables, which can be used to control how the boot loader, or any other software running on the system, behaves. Boot loader environment variables are similar to environment variables that can be set on UNIX or DOS systems.

Environment variables that have values are stored in flash memory outside of the flash file system.

Each line in these files contains an environment variable name and an equal sign followed by the value of the variable. A variable has no value if it is not listed in this file; it has a value if it is listed in the file even if the value is a null string. A variable that is set to a null string (for example, “ ”) is a variable with a value. Many environment variables are predefined and have default values.

Environment variables store two kinds of data:

- Data that controls code, which does not read the Cisco IOS configuration file. For example, the name of a boot loader helper file, which extends or patches the functionality of the boot loader can be stored as an environment variable.
- Data that controls code, which is responsible for reading the Cisco IOS configuration file. For example, the name of the Cisco IOS configuration file can be stored as an environment variable.

You can change the settings of the environment variables by accessing the boot loader or by using Cisco IOS commands. Under normal circumstances, it is not necessary to alter the setting of the environment variables.

Note For complete syntax and usage information for the boot loader commands and environment variables, see the command reference for this release.

Table 3-4 describes the function of the most common environment variables.

Table 3-4 Environment Variables

Variable	Boot Loader Command	Cisco IOS Global Configuration Command
BOOT	**set BOOT** *filesystem***:**/*file-url* ... A semicolon-separated list of executable files to try to load and execute when automatically booting. If the BOOT environment variable is not set, the system attempts to load and execute the first executable image it can find by using a recursive, depth-first search through the flash file system. If the BOOT variable is set but the specified images cannot be loaded, the system attempts to boot the first bootable file that it can find in the flash file system.	**boot system** f*ilesystem:/file-url* ... Specifies the Cisco IOS image to load during the next boot cycle. This command changes the setting of the BOOT environment variable.
MANUAL_BOOT	**set MANUAL_BOOT yes** Decides whether the switch automatically or manually boots. Valid values are 1, yes, 0, and no. If it is set to no or 0, the boot loader attempts to automatically boot the system. If it is set to anything else, you must manually boot the switch from the boot loader mode.	**boot manual** Enables manually booting the switch during the next boot cycle and changes the setting of the MANUAL_BOOT environment variable. The next time you reboot the system, the switch is in boot loader mode. To boot the system, use the **boot flash:***filesystem***:**/*file-url* boot loader command, and specify the name of the bootable image.
CONFIG_FILE	**set CONFIG_FILE flash:**/*file-url* Changes the filename that Cisco IOS uses to read and write a nonvolatile copy of the system configuration.	**boot config-file flash:**/*file-url* Specifies the filename that Cisco IOS uses to read and write a nonvolatile copy of the system configuration. This command changes the CONFIG_FILE environment variable.

Scheduling a Reload of the Software Image

You can schedule a reload of the software image to occur on the switch at a later time (for example, late at night or during the weekend when the switch is used less), or you can synchronize a reload network-wide (for example, to perform a software upgrade on all switches in the network).

Note A scheduled reload must take place within approximately 24 days.

Configuring a Scheduled Reload

To configure your switch to reload the software image at a later time, use one of these commands in privileged EXEC mode:

- **reload in** [*hh***:**]*mm* [*text*]

 This command schedules a reload of the software to take affect in the specified minutes or hours and minutes. The reload must take place within approximately 24 days. You can specify the reason for the reload in a string up to 255 characters in length.

- **reload at** *hh***:***mm* [*month day* | *day month*] [*text*]

 This command schedules a reload of the software to take place at the specified time (using a 24-hour clock). If you specify the month and day, the reload is scheduled to take place at the specified time and date. If you do not specify the month and day, the reload takes place at the specified time on the current day (if the specified time is later than the current time) or on the next day (if the specified time is earlier than the current time). Specifying 00:00 schedules the reload for midnight.

Note Use the **at** keyword only if the switch system clock has been set (through Network Time Protocol (NTP), the hardware calendar, or manually). The time is relative to the configured time zone on the switch. To schedule reloads across several switches to occur simultaneously, the time on each switch must be synchronized with NTP.

The **reload** command halts the system. If the system is not set to manually boot, it reboots itself. Use the **reload** command after you save the switch configuration information to the startup configuration (**copy running-config startup-config**).

If your switch is configured for manual booting, do not reload it from a virtual terminal. This restriction prevents the switch from entering the boot loader mode and thereby taking it from the remote user's control.

If you modify your configuration file, the switch prompts you to save the configuration before reloading. During the save operation, the system requests whether you want to proceed with the save if the CONFIG_FILE environment variable points to a startup configuration file that no longer exists. If you proceed in this situation, the system enters setup mode upon reload.

This example shows how to reload the software on the switch on the current day at 7:30 p.m:

```
Switch# reload at 19:30
Reload scheduled for 19:30:00 UTC Wed Jun 5 1996 (in 2 hours and 25 minutes)
Proceed with reload? [confirm]
```

This example shows how to reload the software on the switch at a future time:

```
Switch# reload at 02:00 jun 20
Reload scheduled for 02:00:00 UTC Thu Jun 20 1996 (in 344 hours and 53 minutes)
Proceed with reload? [confirm]
```

To cancel a previously scheduled reload, use the **reload cancel** privileged EXEC command.

Displaying Scheduled Reload Information

To display information about a previously scheduled reload or to find out if a reload has been scheduled on the switch, use the **show reload** privileged EXEC command.

It displays reload information including the time the reload is scheduled to occur and the reason for the reload (if it was specified when the reload was scheduled).



Configuring Cisco IOS CNS Agents

This chapter describes how to configure the Cisco IOS CNS agents on the Cisco ME 3400 switch.

Note For complete configuration information for the Cisco Configuration Engine, see this URL on Cisco.com http://www.cisco.com/en/US/products/sw/netmgtsw/ps4617/tsd_products_support_series_home.html

This chapter consists of these sections:

- Understanding Cisco Configuration Engine Software, page 4-1
- Understanding Cisco IOS Agents, page 4-5
- Configuring Cisco IOS Agents, page 4-6
- Displaying CNS Configuration, page 4-12

Understanding Cisco Configuration Engine Software

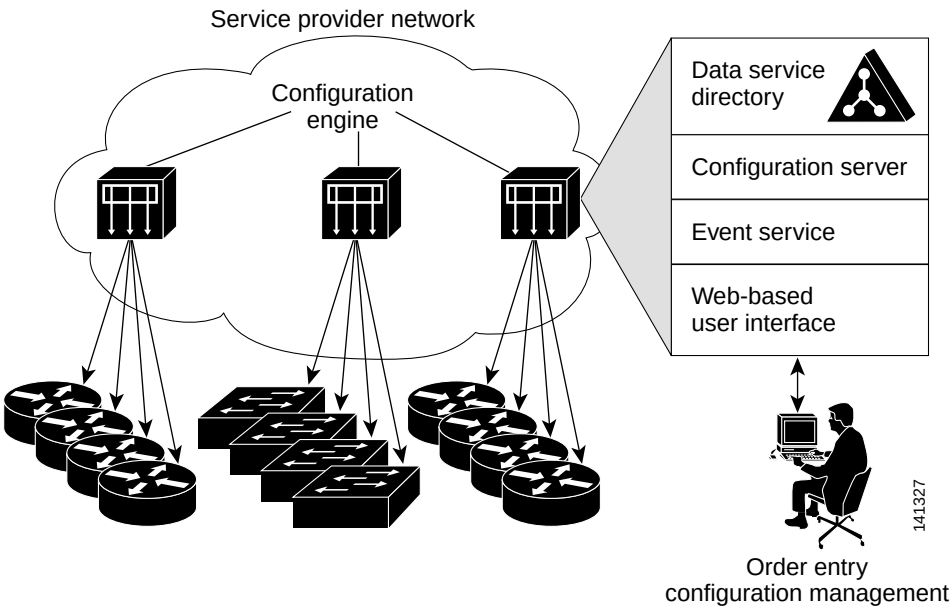
The Cisco Configuration Engine is network management software that acts as a configuration service for automating the deployment and management of network devices and services (see Figure 4-1). Each Configuration Engine manages a group of Cisco devices (switches and routers) and the services that they deliver, storing their configurations and delivering them as needed. The Configuration Engine automates initial configurations and configuration updates by generating device-specific configuration changes, sending them to the device, executing the configuration change, and logging the results.

The Configuration Engine supports standalone and server modes and has these CNS components:

- Configuration service (web server, file manager, and namespace mapping server)
- Event service (event gateway)
- Data service directory (data models and schema)

In standalone mode, the Configuration Engine supports an embedded Directory Service. In this mode, no external directory or other data store is required. In server mode, the Configuration Engine supports the use of a user-defined external directory.

Figure 4-1 Configuration Engine Architectural Overview



These sections contain this conceptual information:

- Configuration Service, page 4-2
- Event Service, page 4-3
- What You Should Know About the CNS IDs and Device Hostnames, page 4-3

Configuration Service

The Configuration Service is the core component of the Cisco Configuration Engine. It consists of a configuration server that works with Cisco IOS CNS agents on the switch. The Configuration Service delivers device and service configurations to the switch for initial configuration and mass reconfiguration by logical groups. Switches receive their initial configuration from the Configuration Service when they start up on the network for the first time.

The Configuration Service uses the CNS Event Service to send and receive configuration change events and to send success and failure notifications.

The configuration server is a web server that uses configuration templates and the device-specific configuration information stored in the embedded (standalone mode) or remote (server mode) directory.

Configuration templates are text files containing static configuration information in the form of CLI commands. In the templates, variables are specified using lightweight directory access protocol (LDAP) URLs that reference the device-specific configuration information stored in a directory.

The Cisco IOS agent can perform a syntax check on received configuration files and publish events to show the success or failure of the syntax check. The configuration agent can either apply configurations immediately or delay the application until receipt of a synchronization event from the configuration server.

Event Service

The Cisco Configuration Engine uses the Event Service for receipt and generation of configuration events. The event agent is on the switch and facilitates the communication between the switch and the event gateway on the Configuration Engine.

The Event Service is a highly capable publish-and-subscribe communication method. The Event Service uses subject-based addressing to send messages to their destinations. Subject-based addressing conventions define a simple, uniform namespace for messages and their destinations.

NameSpace Mapper

The Configuration Engine includes the NameSpace Mapper (NSM) that provides a lookup service for managing logical groups of devices based on application, device or group ID, and event.

Cisco IOS devices recognize only event subject-names that match those configured in Cisco IOS software; for example, cisco.cns.config.load. You can use the namespace mapping service to designate events by using any desired naming convention. When you have populated your data store with your subject names, NSM changes your event subject-name strings to those known by Cisco IOS.

For a subscriber, when given a unique device ID and event, the namespace mapping service returns a set of events to which to subscribe. Similarly, for a publisher, when given a unique group ID, device ID, and event, the mapping service returns a set of events on which to publish.

What You Should Know About the CNS IDs and Device Hostnames

The Cisco Configuration Engine assumes that a unique identifier is associated with each configured switch. This unique identifier can take on multiple synonyms, where each synonym is unique within a particular namespace. The event service uses namespace content for subject-based addressing of messages.

The Configuration Engine intersects two namespaces, one for the event bus and the other for the configuration server. Within the scope of the configuration server namespace, the term *ConfigID* is the unique identifier for a device. Within the scope of the event bus namespace, the term *DeviceID* is the CNS unique identifier for a device.

Because the Configuration Engine uses both the event bus and the configuration server to provide configurations to devices, you must define both ConfigID and Device ID for each configured switch.

Within the scope of a single instance of the configuration server, no two configured switches can share the same value for ConfigID. Within the scope of a single instance of the event bus, no two configured switches can share the same value for DeviceID.

ConfigID

Each configured switch has a unique ConfigID, which serves as the key into the Configuration Engine directory for the corresponding set of switch CLI attributes. The ConfigID defined on the switch must match the ConfigID for the corresponding switch definition on the Configuration Engine.

The ConfigID is fixed at startup time and cannot be changed until the device restarts, even if the switch hostname is reconfigured.

DeviceID

Each configured switch participating on the event bus has a unique DeviceID, which is analogous to the switch source address so that the switch can be targeted as a specific destination on the bus. All switches configured with the **cns config partial** global configuration command must access the event bus. Therefore, the DeviceID, as originated on the switch, must match the DeviceID of the corresponding switch definition in the Configuration Engine.

The origin of the DeviceID is defined by the Cisco IOS hostname of the switch. However, the DeviceID variable and its usage reside within the event gateway adjacent to the switch.

The logical Cisco IOS termination point on the event bus is embedded in the event gateway, which in turn functions as a proxy on behalf of the switch. The event gateway represents the switch and its corresponding DeviceID to the event bus.

The switch declares its hostname to the event gateway immediately after the successful connection to the event gateway. The event gateway couples the DeviceID value to the Cisco IOS hostname each time this connection is established. The event gateway caches this DeviceID value for the duration of its connection to the switch.

Hostname and DeviceID

The DeviceID is fixed at the time of the connection to the event gateway and does not change even when the switch hostname is reconfigured.

When changing the switch hostname on the switch, the only way to refresh the DeviceID is to break the connection between the switch and the event gateway. Enter the **no cns event** global configuration command followed by the **cns event** global configuration command.

When the connection is re-established, the switch sends its modified hostname to the event gateway. The event gateway redefines the DeviceID to the new value.

Caution When using the Configuration Engine user interface, you must first set the DeviceID field to the hostname value that the switch acquires *after*–not *before*–you use the **cns config initial** global configuration command at the switch. Otherwise, subsequent **cns config partial** global configuration command operations malfunction.

Using Hostname, DeviceID, and ConfigID

In standalone mode, when a hostname value is set for a switch, the configuration server uses the hostname as the DeviceID when an event is sent on hostname. If the hostname has not been set, the event is sent on the cn=<*value*> of the device.

In server mode, the hostname is not used. In this mode, the unique DeviceID attribute is always used for sending an event on the bus. If this attribute is not set, you cannot update the switch.

These and other associated attributes (tag value pairs) are set when you run **Setup** on the Configuration Engine.

Note For more information about running the setup program on the Configuration Engine, see the Configuration Engine setup and configuration guide at this URL on cisco.com: http://www.cisco.com/en/US/products/sw/netmgtsw/ps4617/products_installation_and_configuration_guide_book09186a00803b59db.html

Understanding Cisco IOS Agents

The CNS event agent feature allows the switch to publish and subscribe to events on the event bus and works with the Cisco IOS agent. The Cisco IOS agent feature supports the switch by providing these features:

- Initial Configuration, page 4-5
- Incremental (Partial) Configuration, page 4-6
- Synchronized Configuration, page 4-6

Initial Configuration

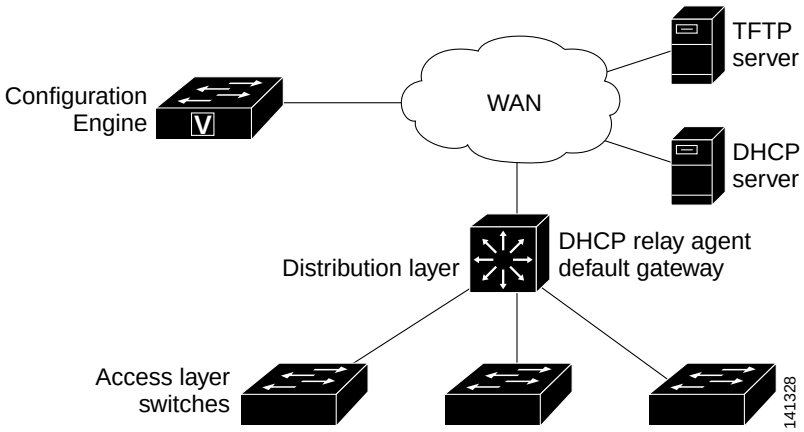
When the switch first comes up, it attempts to get an IP address by broadcasting a DHCP request on the network. Assuming there is no DHCP server on the subnet, the distribution switch acts as a DHCP relay agent and forwards the request to the DHCP server. Upon receiving the request, the DHCP server assigns an IP address to the new switch and includes the TFTP server IP address, the path to the bootstrap configuration file, and the default gateway IP address in a unicast reply to the DHCP relay agent. The DHCP relay agent forwards the reply to the switch.

The switch automatically configures the assigned IP address on interface VLAN 1 (the default) and downloads the bootstrap configuration file from the TFTP server. Upon successful download of the bootstrap configuration file, the switch loads the file in its running configuration.

The Cisco IOS agents initiate communication with the Configuration Engine by using the appropriate ConfigID and EventID. The Configuration Engine maps the Config ID to a template and downloads the full configuration file to the switch.

Figure 4-2 shows a sample network configuration for retrieving the initial bootstrap configuration file by using DHCP-based autoconfiguration.

Figure 4-2 Initial Configuration Overview



Incremental (Partial) Configuration

After the network is running, new services can be added by using the Cisco IOS agent. Incremental (partial) configurations can be sent to the switch. The actual configuration can be sent as an event payload by way of the event gateway (push operation) or as a signal event that triggers the switch to initiate a pull operation.

The switch can check the syntax of the configuration before applying it. If the syntax is correct, the switch applies the incremental configuration and publishes an event that signals success to the configuration server. If the switch does not apply the incremental configuration, it publishes an event showing an error status. When the switch has applied the incremental configuration, it can write it to NVRAM or wait until signaled to do so.

Synchronized Configuration

When the switch receives a configuration, it can defer application of the configuration upon receipt of a write-signal event. The write-signal event tells the switch not to save the updated configuration into its NVRAM. The switch uses the updated configuration as its running configuration. This ensures that the switch configuration is synchronized with other network activities before saving the configuration in NVRAM for use at the next reboot.

Configuring Cisco IOS Agents

The Cisco IOS agents embedded in the switch Cisco IOS software allow the switch to be connected and automatically configured as described in the "Enabling Automated CNS Configuration" section on page 4-6. If you want to change the configuration or install a custom configuration, see these sections for instructions:

- Enabling the CNS Event Agent, page 4-8
- Enabling the Cisco IOS CNS Agent, page 4-9

Enabling Automated CNS Configuration

To enable automated CNS configuration of the switch, you must first complete the prerequisites in Table 4-1. When you complete them, power on the switch. At the **setup** prompt, do nothing: The switch begins the initial configuration as described in the "Initial Configuration" section on page 4-5. When the full configuration file is loaded on your switch, you need to do nothing else.

Table 4-1 Prerequisites for Enabling Automatic Configuration

Device	Required Configuration
Access switch	Factory default (no configuration file)
Distribution switch	• IP helper address • Enable DHCP relay agent • IP routing (if used as default gateway)
DHCP server	• IP address assignment • TFTP server IP address • Path to bootstrap configuration file on the TFTP server • Default gateway IP address
TFTP server	• A bootstrap configuration file that includes the CNS configuration commands that enable the switch to communicate with the Configuration Engine • The switch configured to use either the switch MAC address or the serial number (instead of the default hostname) to generate the ConfigID and EventID • The CNS event agent configured to push the configuration file to the switch
CNS Configuration Engine	One or more templates for each type of device, with the ConfigID of the device mapped to the template.

Note For more information about running the setup program and creating templates on the Configuration Engine, see the *Cisco Configuration Engine Installation and Setup Guide, 1.5 for Linux* at this URL: http://www.cisco.com/en/US/products/sw/netmgtsw/ps4617/products_installation_and_configuration_guide_book09186a00803b59db.html

Enabling the CNS Event Agent

Note You must enable the CNS event agent on the switch before you enable the CNS configuration agent.

Beginning in privileged EXEC mode, follow these steps to enable the CNS event agent on the switch:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**cns event** {*ip-address* \| *hostname*} [*port-number*] [**backup**] [**init-retry** *retry-count*] [**keepalive** *seconds retry-count*] [**source** *ip-address*]	Enable the event agent, and enter the gateway parameters. • For {*ip-address* \| *hostname*}, enter either the IP address or the hostname of the event gateway. • (Optional) For *port number*, enter the port number for the event gateway. The default port number is 11011. • (Optional) Enter **backup** to show that this is the backup gateway. (If omitted, this is the primary gateway.) • (Optional) For **init-retry** *retry-count*, enter the number of initial retries before switching to backup. The default is 3. • (Optional) For **keepalive** *seconds*, enter how often the switch sends keepalive messages. For *retry-count*, enter the number of unanswered keepalive messages that the switch sends before the connection is terminated. The default for each is 0. • (Optional) For **source** *ip-address*, enter the source IP address of this device. **Note** Though visible in the command-line help string, the **encrypt** and **force-fmt1** keywords are not supported.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show cns event connections**	Verify information about the event agent.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable the CNS event agent, use the **no cns event** {*ip-address* | *hostname*} global configuration command.

This example shows how to enable the CNS event agent, set the IP address gateway to 10.180.1.27, set 120 seconds as the keepalive interval, and set 10 as the retry count.

```
Switch(config)# cns event 10.180.1.27 keepalive 120 10
```

Enabling the Cisco IOS CNS Agent

After enabling the CNS event agent, start the Cisco IOS CNS agent on the switch. You can enable the Cisco IOS agent with these commands:

- The **cns config initial** global configuration command enables the Cisco IOS agent and initiates an initial configuration on the switch.
- The **cns config partial** global configuration command enables the Cisco IOS agent and initiates a partial configuration on the switch. You can then use the Configuration Engine to remotely send incremental configurations to the switch.

Enabling an Initial Configuration

Beginning in privileged EXEC mode, follow these steps to enable the CNS configuration agent and initiate an initial configuration on the switch:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**cns config connect-intf** *interface-prefix* [**ping-interval** *seconds*] [**retries** *num*]	Enter the connect-interface-config submode, and specify the interface for connecting to the Configuration Engine. • Enter the *interface-prefix* for the connecting interface. You must specify the interface type but need not specify the interface number. • (Optional) For **ping-interval** *seconds*, enter the interval between successive ping attempts. The range is 1 to 30 seconds. The default is 10 seconds. • (Optional) For **retries** *num*, enter the number of ping retries. The range is 1 to 30. The default is 5.
Step 3	**config-cli** or **line-cli**	Enter **config-cli** to connect to the Configuration Engine through the interface defined in **cns config connect-intf**. Enter **line-cli** to connect to the Configuration Engine through modem dialup lines. **Note** The **config-cli** interface configuration command accepts the special character **&** that acts as a placeholder for the interface name. When the configuration is applied, the **&** is replaced with the interface name. For example, to connect through FastEthernet0/1, the command `config-cli ip route 0.0.0.0 0.0.0.0 &` generates the command `ip route 0.0.0.0 0.0.0.0 FastEthernet0/1`.
Step 4	**exit**	Return to global configuration mode.
Step 5	**hostname** *name*	Enter the hostname for the switch.
Step 6	**ip route** *network-number*	Establish a static route to the Configuration Engine whose IP address is *network-number.*

	Command	Purpose
Step 7	**cns id** *interface num* {**dns-reverse** \| **ipaddress** \| **mac-address**} [**event**] or **cns id** {**hardware-serial** \| **hostname** \| **string** *string*} [**event**]	Set the unique EventID or ConfigID used by the Configuration Engine. • For *interface num*, enter the type of interface–for example, Ethernet, Group-Async, Loopback, or Virtual-Template. This setting specifies from which interface the IP or MAC address should be retrieved to define the unique ID. • For {**dns-reverse** \| **ipaddress** \| **mac-address**} enter **dns-reverse** to retrieve the hostname and assign it as the unique ID, enter **ipaddress** to use the IP address, or enter **mac-address** to use the MAC address as the unique ID. • (Optional) Enter **event** to set the ID to be the event-id value used to identify the switch. • For {**hardware-serial** \| **hostname\| string** *string*}, enter **hardware-serial** to set the switch serial number as the unique ID, enter **hostname** (the default) to select the switch hostname as the unique ID, or enter an arbitrary text string for **string** *string* as the unique ID.
Step 8	**cns config initial** {*ip-address* \| *hostname*} [*port-number*] [**event**] [**no-persist**] [**page** *page*] [**source** *ip-address*] [**syntax-check**]	Enable the Cisco IOS agent, and initiate an initial configuration. • For {*ip-address* \| *hostname*}, enter the IP address or the hostname of the configuration server. • (Optional) For *port-number*, enter the port number of the configuration server. The default port number is 80. • (Optional) Enable **event** for configuration success, failure, or warning messages when the configuration is finished. • (Optional) Enable **no-persist** to suppress the automatic writing to NVRAM of the configuration pulled as a result of entering the **cns config initial** global configuration command. If the **no-persist** keyword is not entered, using the **cns config initial** command causes the resultant configuration to be automatically written to NVRAM. • (Optional) For **page** *page*, enter the web page of the initial configuration. The default is /Config/config/asp. • (Optional) Enter **source** *ip-address* to use for source IP address. • (Optional) Enable **syntax-check** to check the syntax when this parameter is entered. **Note** Though visible in the command-line help string, the **encrypt** keyword is not supported.
Step 9	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 10	**show cns config connections**	Verify information about the configuration agent.
Step 11	**show running-config**	Verify your entries.

To disable the CNS Cisco IOS agent, use the **no cns config initial** {*ip-address* | *hostname*} global configuration command.

This example shows how to configure an initial configuration on a remote switch. The switch hostname is the unique ID. The Cisco Configuration Engine IP address is 172.28.129.22.

```
Switch(config)# cns config connect-intf serial ping-interval 1 retries 1
Switch(config-cns-conn-if)# config-cli ip address negotiated
Switch(config-cns-conn-if)# config-cli encapsulation ppp
Switch(config-cns-conn-if)# config-cli ip directed-broadcast
Switch(config-cns-conn-if)# config-cli no keepalive
Switch(config-cns-conn-if)# config-cli no shutdown
Switch(config-cns-conn-if)# exit
Switch(config)# hostname RemoteSwitch
RemoteSwitch(config)# ip route 10.1.1.1 255.255.255.255 11.11.11.1
RemoteSwitch(config)# cns id Ethernet 0 ipaddress
RemoteSwitch(config)# cns config initial 10.1.1.1 no-persist
```

Enabling a Partial Configuration

Beginning in privileged EXEC mode, follow these steps to enable the Cisco IOS agent and to initiate a partial configuration on the switch:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**cns config partial** {*ip-address* \| *hostname*} [*port-number*] [**source** *ip-address*]	Enable the configuration agent, and initiate a partial configuration. • For {*ip-address* \| *hostname*}, enter the IP address or the hostname of the configuration server. • (Optional) For *port-number*, enter the port number of the configuration server. The default port number is 80. • (Optional) Enter **source** *ip-address* to use for the source IP address. **Note** Though visible in the command-line help string, the **encrypt** keyword is not supported.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show cns config stats** or **show cns config outstanding**	Verify information about the configuration agent.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable the Cisco IOS agent, use the **no cns config partial** {*ip-address* | *hostname*} global configuration command. To cancel a partial configuration, use the **cns config cancel** privileged EXEC command.

Displaying CNS Configuration

You can use the privileged EXEC commands in Table 4-2 to display CNS configuration information.

Table 4-2 Displaying CNS Configuration

Command	Purpose
show cns config connections	Displays the status of the CNS Cisco IOS agent connections.
show cns config outstanding	Displays information about incremental (partial) CNS configurations that have started but are not yet completed.
show cns config stats	Displays statistics about the Cisco IOS agent.
show cns event connections	Displays the status of the CNS event agent connections.
show cns event stats	Displays statistics about the CNS event agent.
show cns event subject	Displays a list of event agent subjects that are subscribed to by applications.



Administering the Switch

This chapter describes how to perform one-time operations to administer the Cisco ME 3400 Ethernet Access switch.

This chapter consists of these sections:

- Managing the System Time and Date, page 5-1
- Configuring a System Name and Prompt, page 5-14
- Creating a Banner, page 5-17
- Managing the MAC Address Table, page 5-19
- Managing the ARP Table, page 5-28

Managing the System Time and Date

You can manage the system time and date on your switch using automatic configuration, such as the Network Time Protocol (NTP), or manual configuration methods.

Note For complete syntax and usage information for the commands used in this section, see the *Cisco IOS Configuration Fundamentals Command Reference, Release 12.2*.

These sections contain this configuration information:

- Understanding the System Clock, page 5-2
- Understanding Network Time Protocol, page 5-2
- Configuring NTP, page 5-4
- Configuring Time and Date Manually, page 5-11

Understanding the System Clock

The heart of the time service is the system clock. This clock runs from the moment the system starts up and keeps track of the date and time.

The system clock can then be set from these sources:

- NTP
- Manual configuration

The system clock can provide time to these services:

- User **show** commands
- Logging and debugging messages

The system clock keeps track of time internally based on Universal Time Coordinated (UTC), also known as Greenwich Mean Time (GMT). You can configure information about the local time zone and summer time (daylight saving time) so that the time appears correctly for the local time zone.

The system clock keeps track of whether the time is *authoritative* or not (that is, whether it has been set by a time source considered to be authoritative). If it is not authoritative, the time is available only for display purposes and is not redistributed. For configuration information, see the "Configuring Time and Date Manually" section on page 5-11.

Understanding Network Time Protocol

The NTP is designed to time-synchronize a network of devices. NTP runs over User Datagram Protocol (UDP), which runs over IP. NTP is documented in RFC 1305.

An NTP network usually gets its time from an authoritative time source, such as a radio clock or an atomic clock attached to a time server. NTP then distributes this time across the network. NTP is extremely efficient; no more than one packet per minute is necessary to synchronize two devices to within a millisecond of one another.

NTP uses the concept of a *stratum* to describe how many NTP hops away a device is from an authoritative time source. A stratum 1 time server has a radio or atomic clock directly attached, a stratum 2 time server receives its time through NTP from a stratum 1 time server, and so on. A device running NTP automatically chooses as its time source the device with the lowest stratum number with which it communicates through NTP. This strategy effectively builds a self-organizing tree of NTP speakers.

NTP avoids synchronizing to a device whose time might not be accurate by never synchronizing to a device that is not synchronized. NTP also compares the time reported by several devices and does not synchronize to a device whose time is significantly different than the others, even if its stratum is lower.

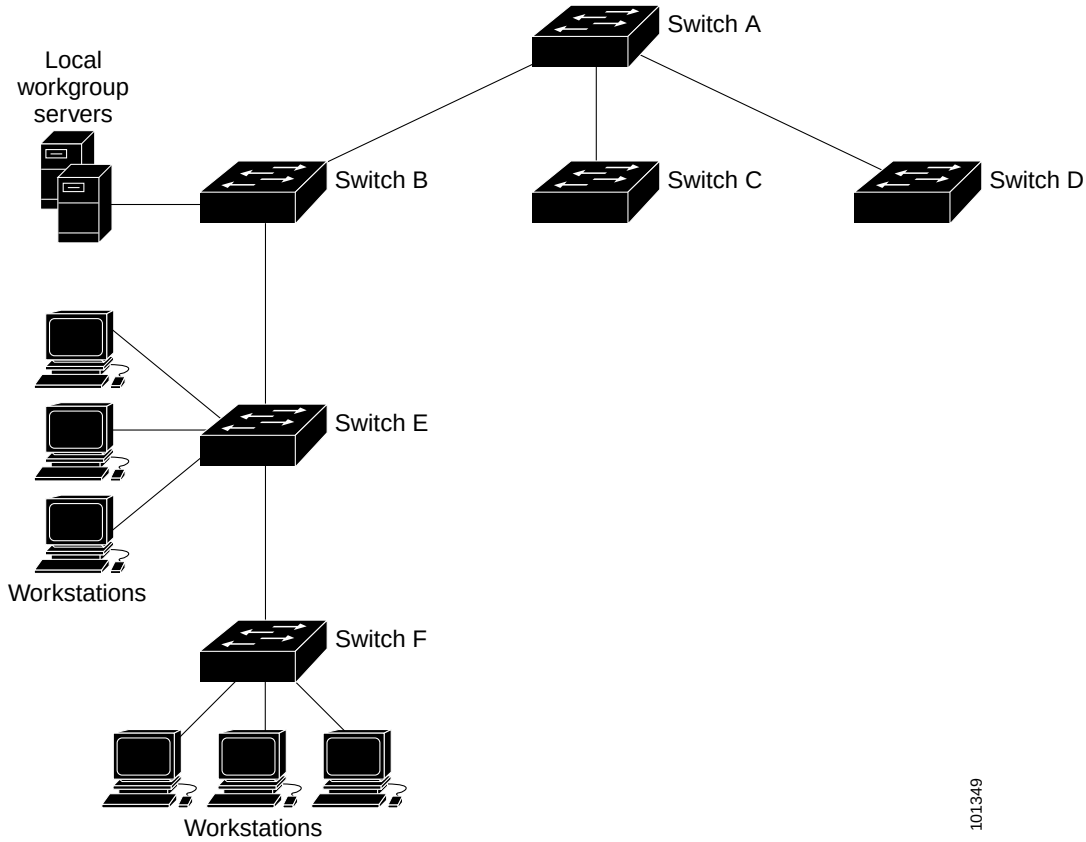
The communications between devices running NTP (known as *associations*) are usually statically configured; each device is given the IP address of all devices with which it should form associations. Accurate timekeeping is possible by exchanging NTP messages between each pair of devices with an association. However, in a LAN environment, NTP can be configured to use IP broadcast messages instead. This alternative reduces configuration complexity because each device can simply be configured to send or receive broadcast messages. However, in that case, information flow is one-way only.

The time kept on a device is a critical resource; you should use the security features of NTP to avoid the accidental or malicious setting of an incorrect time. Two mechanisms are available: an access list-based restriction scheme and an encrypted authentication mechanism.

Cisco's implementation of NTP does not support stratum 1 service; it is not possible to connect to a radio or atomic clock. We recommend that the time service for your network be derived from the public NTP servers available on the IP Internet.

Figure 5-1 shows a typical network example using NTP. Switch A is the NTP master, with Switches B, C, and D configured in NTP server mode, in server association with Switch A. Switch E is configured as an NTP peer to the upstream and downstream switches, Switch B and Switch F.

Figure 5-1 Typical NTP Network Configuration



If the network is isolated from the Internet, Cisco's implementation of NTP allows a device to act as if it is synchronized through NTP, when in fact it has learned the time by using other means. Other devices then synchronize to that device through NTP.

When multiple sources of time are available, NTP is always considered to be more authoritative. NTP time overrides the time set by any other method.

Several manufacturers include NTP software for their host systems, and a publicly available version for systems running UNIX and its various derivatives is also available. This software allows host systems to be time-synchronized as well.

Configuring NTP

The switch does not have a hardware-supported clock and cannot function as an NTP master clock to which peers synchronize themselves when an external NTP source is not available. The switch also has no hardware support for a calendar. As a result, the **ntp update-calendar** and the **ntp master** global configuration commands are not available.

These sections contain this configuration information:

- Default NTP Configuration, page 5-4
- Configuring NTP Authentication, page 5-5
- Configuring NTP Associations, page 5-6
- Configuring NTP Broadcast Service, page 5-7
- Configuring NTP Access Restrictions, page 5-8
- Configuring the Source IP Address for NTP Packets, page 5-10
- Displaying the NTP Configuration, page 5-11

Default NTP Configuration

Table 5-1 shows the default NTP configuration.

Table 5-1 Default NTP Configuration

Feature	Default Setting
NTP authentication	Disabled. No authentication key is specified.
NTP peer or server associations	None configured.
NTP broadcast service	Disabled; no interface sends or receives NTP broadcast packets.
NTP access restrictions	No access control is specified.
NTP packet source IP address	The source address is set by the outgoing interface.

NTP is enabled on all interfaces by default. All interfaces receive NTP packets.

Configuring NTP Authentication

This procedure must be coordinated with the administrator of the NTP server; the information you configure in this procedure must be matched by the servers used by the switch to synchronize its time to the NTP server.

Beginning in privileged EXEC mode, follow these steps to authenticate the associations (communications between devices running NTP that provide for accurate timekeeping) with other devices for security purposes:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ntp authenticate**	Enable the NTP authentication feature, which is disabled by default.
Step 3	**ntp authentication-key** *number* **md5** *value*	Define the authentication keys. By default, none are defined. • For *number*, specify a key number. The range is 1 to 4294967295. • **md5** specifies that message authentication support is provided by using the message digest algorithm 5 (MD5). • For *value*, enter an arbitrary string of up to eight characters for the key. The switch does not synchronize to a device unless both have one of these authentication keys, and the key number is specified by the **ntp trusted-key** *key-number* command.
Step 4	**ntp trusted-key** *key-number*	Specify one or more key numbers (defined in Step 3) that a peer NTP device must provide in its NTP packets for this switch to synchronize to it. By default, no trusted keys are defined. For *key-number*, specify the key defined in Step 3. This command provides protection against accidentally synchronizing the switch to a device that is not trusted.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable NTP authentication, use the **no ntp authenticate** global configuration command. To remove an authentication key, use the **no ntp authentication-key** *number* global configuration command. To disable authentication of the identity of a device, use the **no ntp trusted-key** *key-number* global configuration command.

This example shows how to configure the switch to synchronize only to devices providing authentication key 42 in the device's NTP packets:

```
Switch(config)# ntp authenticate
Switch(config)# ntp authentication-key 42 md5 aNiceKey
Switch(config)# ntp trusted-key 42
```

Configuring NTP Associations

An NTP association can be a peer association (this switch can either synchronize to the other device or allow the other device to synchronize to it), or it can be a server association (meaning that only this switch synchronizes to the other device, and not the other way around).

Beginning in privileged EXEC mode, follow these steps to form an NTP association with another device:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ntp peer** *ip-address* [**version** *number*] [**key** *keyid*] [**source** *interface*] [**prefer**]	Configure the switch system clock to synchronize a peer or to be synchronized by a peer (peer association).
	or	or
	ntp server *ip-address* [**version** *number*] [**key** *keyid*] [**source** *interface*] [**prefer**]	Configure the switch system clock to be synchronized by a time server (server association). No peer or server associations are defined by default. • For *ip-address* in a peer association, specify either the IP address of the peer providing, or being provided, the clock synchronization. For a server association, specify the IP address of the time server providing the clock synchronization. • (Optional) For *number*, specify the NTP version number. The range is 1 to 3. By default, Version 3 is selected. • (Optional) For *keyid*, enter the authentication key defined with the **ntp authentication-key** global configuration command. • (Optional) For *interface*, specify the interface from which to pick the IP source address. By default, the source IP address is taken from the outgoing interface. • (Optional) Enter the **prefer** keyword to make this peer or server the preferred one that provides synchronization. This keyword reduces switching back and forth between peers and servers.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

You need to configure only one end of an association; the other device can automatically establish the association. If you are using the default NTP version (Version 3) and NTP synchronization does not occur, try using NTP Version 2. Many NTP servers on the Internet run Version 2.

To remove a peer or server association, use the **no ntp peer** *ip-address* or the **no ntp server** *ip-address* global configuration command.

This example shows how to configure the switch to synchronize its system clock with the clock of the peer at IP address 172.16.22.44 using NTP Version 2:

```
Switch(config)# ntp server 172.16.22.44 version 2
```

Configuring NTP Broadcast Service

The communications between devices running NTP (known as *associations*) are usually statically configured; each device is given the IP addresses of all devices with which it should form associations. Accurate timekeeping is possible by exchanging NTP messages between each pair of devices with an association. However, in a LAN environment, NTP can be configured to use IP broadcast messages instead. This alternative reduces configuration complexity because each device can simply be configured to send or receive broadcast messages. However, the information flow is one-way only.

The switch can send or receive NTP broadcast packets on an interface-by-interface basis if there is an NTP broadcast server, such as a router, broadcasting time information on the network. The switch can send NTP broadcast packets to a peer so that the peer can synchronize to it. The switch can also receive NTP broadcast packets to synchronize its own clock. This section provides procedures for both sending and receiving NTP broadcast packets.

Beginning in privileged EXEC mode, follow these steps to configure the switch to send NTP broadcast packets to peers so that they can synchronize their clock to the switch:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to send NTP broadcast packets, and enter interface configuration mode.
Step 3	**no shutdown**	Enable the port, if necessary. By default, user network interfaces (UNIs) are disabled, and network node interfaces (NNIs) are enabled.
Step 4	**ntp broadcast** [**version** *number*] [**key** *keyid*] [*destination-address*]	Enable the interface to send NTP broadcast packets to a peer. By default, this feature is disabled on all interfaces. • (Optional) For *number*, specify the NTP version number. The range is 1 to 3. If you do not specify a version, Version 3 is used. • (Optional) For *keyid*, specify the authentication key to use when sending packets to the peer. • (Optional) For *destination-address*, specify the IP address of the peer that is synchronizing its clock to this switch.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.
Step 8		Configure the connected peers to receive NTP broadcast packets as described in the next procedure.

To disable the interface from sending NTP broadcast packets, use the **no ntp broadcast** interface configuration command.

This example shows how to configure a port to send NTP Version 2 packets:

```
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ntp broadcast version 2
```

Beginning in privileged EXEC mode, follow these steps to configure the switch to receive NTP broadcast packets from connected peers:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to receive NTP broadcast packets, and enter interface configuration mode.
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled, and NNIs are enabled.
Step 4	**ntp broadcast client**	Enable the interface to receive NTP broadcast packets. By default, no interfaces receive NTP broadcast packets.
Step 5	**exit**	Return to global configuration mode.
Step 6	**ntp broadcastdelay** *microseconds*	(Optional) Change the estimated round-trip delay between the switch and the NTP broadcast server. The default is 3000 microseconds; the range is 1 to 999999.
Step 7	**end**	Return to privileged EXEC mode.
Step 8	**show running-config**	Verify your entries.
Step 9	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable an interface from receiving NTP broadcast packets, use the **no ntp broadcast client** interface configuration command. To change the estimated round-trip delay to the default, use the **no ntp broadcastdelay** global configuration command.

This example shows how to configure a port to receive NTP broadcast packets:

```
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ntp broadcast client
```

Configuring NTP Access Restrictions

You can control NTP access on two levels as described in these sections:

- Creating an Access Group and Assigning a Basic IP Access List, page 5-9
- Disabling NTP Services on a Specific Interface, page 5-10

Creating an Access Group and Assigning a Basic IP Access List

Beginning in privileged EXEC mode, follow these steps to control access to NTP services by using access lists:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ntp access-group** {**query-only** \| **serve-only** \| **serve** \| **peer**} *access-list-number*	Create an access group, and apply a basic IP access list. The keywords have these meanings: • **query-only**—Allows only NTP control queries. • **serve-only**—Allows only time requests. • **serve**—Allows time requests and NTP control queries, but does not allow the switch to synchronize to the remote device. • **peer**—Allows time requests and NTP control queries and allows the switch to synchronize to the remote device. For *access-list-number*, enter a standard IP access list number from 1 to 99.
Step 3	**access-list** *access-list-number* **permit** *source* [*source-wildcard*]	Create the access list. • For *access-list-number*, enter the number specified in Step 2. • Enter the **permit** keyword to permit access if the conditions are matched. • For *source*, enter the IP address of the device that is permitted access to the switch. • (Optional) For *source-wildcard*, enter the wildcard bits to be applied to the source. **Note** When creating an access list, remember that, by default, the end of the access list contains an implicit deny statement for everything if it did not find a match before reaching the end.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

The access group keywords are scanned in this order, from least restrictive to most restrictive:

1. **peer**—Allows time requests and NTP control queries and allows the switch to synchronize itself to a device whose address passes the access list criteria.
2. **serve**—Allows time requests and NTP control queries, but does not allow the switch to synchronize itself to a device whose address passes the access list criteria.
3. **serve-only**—Allows only time requests from a device whose address passes the access list criteria.
4. **query-only**—Allows only NTP control queries from a device whose address passes the access list criteria.

If the source IP address matches the access lists for more than one access type, the first type is granted. If no access groups are specified, all access types are granted to all devices. If any access groups are specified, only the specified access types are granted.

To remove access control to the switch NTP services, use the **no ntp access-group** {**query-only** | **serve-only** | **serve** | **peer**} global configuration command.

This example shows how to configure the switch to allow itself to synchronize to a peer from access list 99. However, the switch restricts access to allow only time requests from access list 42:

```
Switch# configure terminal
Switch(config)# ntp access-group peer 99
Switch(config)# ntp access-group serve-only 42
Switch(config)# access-list 99 permit 172.20.130.5
Switch(config)# access list 42 permit 172.20.130.6
```

Disabling NTP Services on a Specific Interface

NTP services are enabled on all interfaces by default.

Beginning in privileged EXEC mode, follow these steps to disable NTP packets from being received on an interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and specify the interface to disable.
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled, and NNIs are enabled.
Step 4	**ntp disable**	Disable NTP packets from being received on the interface. By default, all interfaces receive NTP packets.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To re-enable receipt of NTP packets on an interface, use the **no ntp disable** interface configuration command.

Configuring the Source IP Address for NTP Packets

When the switch sends an NTP packet, the source IP address is normally set to the address of the interface through which the NTP packet is sent. Use the **ntp source** global configuration command when you want to use a particular source IP address for all NTP packets. The address is taken from the specified interface. This command is useful if the address on an interface cannot be used as the destination for reply packets.

Beginning in privileged EXEC mode, follow these steps to configure a specific interface from which the IP source address is to be taken:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ntp source** *type number*	Specify the interface type and number from which the IP source address is taken. By default, the source address is set by the outgoing interface.
Step 3	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

The specified interface is used for the source address for all packets sent to all destinations. If a source address is to be used for a specific association, use the **source** keyword in the **ntp peer** or **ntp server** global configuration command as described in the "Configuring NTP Associations" section on page 5-6.

Displaying the NTP Configuration

You can use two privileged EXEC commands to display NTP information:

- **show ntp associations** [**detail**]
- **show ntp status**

For detailed information about the fields in these displays, see the *Cisco IOS Configuration Fundamentals Command Reference, Release 12.2*.

Configuring Time and Date Manually

If no other source of time is available, you can manually configure the time and date after the system is restarted. The time remains accurate until the next system restart. We recommend that you use manual configuration only as a last resort. If you have an outside source to which the switch can synchronize, you do not need to manually set the system clock.

These sections contain this configuration information:

- Setting the System Clock, page 5-11
- Displaying the Time and Date Configuration, page 5-12
- Configuring the Time Zone, page 5-12
- Configuring Summer Time (Daylight Saving Time), page 5-13

Setting the System Clock

If you have an outside source on the network that provides time services, such as an NTP server, you do not need to manually set the system clock.

Beginning in privileged EXEC mode, follow these steps to set the system clock:

	Command	Purpose
Step 1	**clock set** *hh*:*mm*:*ss day month year* or **clock set** *hh*:*mm*:*ss month day year*	Manually set the system clock using one of these formats. • For *hh*:*mm*:*ss*, specify the time in hours (24-hour format), minutes, and seconds. The time specified is relative to the configured time zone. • For *day*, specify the day by date in the month. • For *month*, specify the month by name. • For *year*, specify the year (no abbreviation).

This example shows how to manually set the system clock to 1:32 p.m. on July 23, 2001:

```
Switch# clock set 13:32:00 23 July 2001
```

Displaying the Time and Date Configuration

To display the time and date configuration, use the **show clock** [**detail**] privileged EXEC command.

The system clock keeps an *authoritative* flag that shows whether the time is authoritative (believed to be accurate). If the system clock has been set by a timing source such as NTP, the flag is set. If the time is not authoritative, it is used only for display purposes. Until the clock is authoritative and the *authoritative* flag is set, the flag prevents peers from synchronizing to the clock when the peers' time is invalid.

The symbol that precedes the **show clock** display has this meaning:

- *—Time is not authoritative.
- (blank)—Time is authoritative.
- .—Time is authoritative, but NTP is not synchronized.

Configuring the Time Zone

Beginning in privileged EXEC mode, follow these steps to manually configure the time zone:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**clock timezone** *zone hours-offset* [*minutes-offset*]	Set the time zone. The switch keeps internal time in universal time coordinated (UTC), so this command is used only for display purposes and when the time is manually set. • For *zone*, enter the name of the time zone to be displayed when standard time is in effect. The default is UTC. • For *hours-offset*, enter the hours offset from UTC. • (Optional) For *minutes-offset*, enter the minutes offset from UTC.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

The *minutes-offset* variable in the **clock timezone** global configuration command is available for those cases where a local time zone is a percentage of an hour different from UTC. For example, the time zone for some sections of Atlantic Canada (AST) is UTC-3.5, where the 3 means 3 hours and .5 means 50 percent. In this case, the necessary command is **clock timezone AST -3 30**.

To set the time to UTC, use the **no clock timezone** global configuration command.

Configuring Summer Time (Daylight Saving Time)

Beginning in privileged EXEC mode, follow these steps to configure summer time (daylight saving time) in areas where it starts and ends on a particular day of the week each year:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**clock summer-time** *zone* **recurring** [*week day month hh*:*mm week day month hh*:*mm* [*offset*]]	Configure summer time to start and end on the specified days every year. Summer time is disabled by default. If you specify **clock summer-time** *zone* **recurring** without parameters, the summer time rules default to the United States rules. • For *zone*, specify the name of the time zone (for example, PDT) to be displayed when summer time is in effect. • (Optional) For *week*, specify the week of the month (1 to 5 or **last**). • (Optional) For *day*, specify the day of the week (Sunday, Monday...). • (Optional) For *month*, specify the month (January, February...). • (Optional) For *hh*:*mm*, specify the time (24-hour format) in hours and minutes. • (Optional) For *offset*, specify the number of minutes to add during summer time. The default is 60.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

The first part of the **clock summer-time** global configuration command specifies when summer time begins, and the second part specifies when it ends. All times are relative to the local time zone. The start time is relative to standard time. The end time is relative to summer time. If the starting month is after the ending month, the system assumes that you are in the southern hemisphere.

This example shows how to specify that summer time starts on the first Sunday in April at 02:00 and ends on the last Sunday in October at 02:00:

```
Switch(config)# clock summer-time PDT recurring 1 Sunday April 2:00 last Sunday October
2:00
```

Beginning in privileged EXEC mode, follow these steps if summer time in your area does not follow a recurring pattern (configure the exact date and time of the next summer time events):

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**clock summer-time** *zone* **date** [*month date year hh*:*mm month date year hh*:*mm* [*offset*]] or **clock summer-time** *zone* **date** [*date month year hh*:*mm date month year hh*:*mm* [*offset*]]	Configure summer time to start on the first date and end on the second date. Summer time is disabled by default. • For *zone*, specify the name of the time zone (for example, PDT) to be displayed when summer time is in effect. • (Optional) For *week*, specify the week of the month (1 to 5 or **last**). • (Optional) For *day*, specify the day of the week (Sunday, Monday...). • (Optional) For *month*, specify the month (January, February...). • (Optional) For *hh*:*mm*, specify the time (24-hour format) in hours and minutes. • (Optional) For *offset*, specify the number of minutes to add during summer time. The default is 60.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

The first part of the **clock summer-time** global configuration command specifies when summer time begins, and the second part specifies when it ends. All times are relative to the local time zone. The start time is relative to standard time. The end time is relative to summer time. If the starting month is after the ending month, the system assumes that you are in the southern hemisphere.

To disable summer time, use the **no clock summer-time** global configuration command.

This example shows how to set summer time to start on October 12, 2000, at 02:00, and end on April 26, 2001, at 02:00:

```
Switch(config)# clock summer-time pdt date 12 October 2000 2:00 26 April 2001 2:00
```

Configuring a System Name and Prompt

You configure the system name on the switch to identify it. By default, the system name and prompt are *Switch*.

If you have not configured a system prompt, the first 20 characters of the system name are used as the system prompt. A greater-than symbol [>] is appended. The prompt is updated whenever the system name changes.

For complete syntax and usage information for the commands used in this section, see the *Cisco IOS Configuration Fundamentals Command Reference, Release 12.2* and the *Cisco IOS IP Command Reference, Volume 2 of 3: Routing Protocols, Release 12.2*.

These sections contain this configuration information:

- Default System Name and Prompt Configuration, page 5-15
- Configuring a System Name, page 5-15
- Understanding DNS, page 5-15

Default System Name and Prompt Configuration

The default switch system name and prompt is *Switch*.

Configuring a System Name

Beginning in privileged EXEC mode, follow these steps to manually configure a system name:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**hostname** *name*	Manually configure a system name. The default setting is *switch*. The name must follow the rules for ARPANET hostnames. They must start with a letter, end with a letter or digit, and have as interior characters only letters, digits, and hyphens. Names can be up to 63 characters.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

When you set the system name, it is also used as the system prompt.

To return to the default hostname, use the **no hostname** global configuration command.

Understanding DNS

The DNS protocol controls the Domain Name System (DNS), a distributed database with which you can map hostnames to IP addresses. When you configure DNS on your switch, you can substitute the hostname for the IP address with all IP commands, such as **ping**, **telnet**, **connect**, and related Telnet support operations.

IP defines a hierarchical naming scheme that allows a device to be identified by its location or domain. Domain names are pieced together with periods (.) as the delimiting characters. For example, Cisco Systems is a commercial organization that IP identifies by a *com* domain name, so its domain name is *cisco.com*. A specific device in this domain, for example, the File Transfer Protocol (FTP) system is identified as *ftp.cisco.com*.

To keep track of domain names, IP has defined the concept of a domain name server, which holds a cache (or database) of names mapped to IP addresses. To map domain names to IP addresses, you must first identify the hostnames, specify the name server that is present on your network, and enable the DNS.

These sections contain this configuration information:

- Default DNS Configuration, page 5-16
- Setting Up DNS, page 5-16
- Displaying the DNS Configuration, page 5-17

Default DNS Configuration

Table 5-2 shows the default DNS configuration.

Table 5-2 Default DNS Configuration

Feature	Default Setting
DNS enable state	Enabled.
DNS default domain name	None configured.
DNS servers	No name server addresses are configured.

Setting Up DNS

Beginning in privileged EXEC mode, follow these steps to set up your switch to use the DNS:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip domain-name** *name*	Define a default domain name that the software uses to complete unqualified hostnames (names without a dotted-decimal domain name). Do not include the initial period that separates an unqualified name from the domain name. At boot time, no domain name is configured; however, if the switch configuration comes from a BOOTP or Dynamic Host Configuration Protocol (DHCP) server, then the default domain name might be set by the BOOTP or DHCP server (if the servers were configured with this information).
Step 3	**ip name-server** *server-address1* [*server-address2 ... server-address6*]	Specify the address of one or more name servers to use for name and address resolution. You can specify up to six name servers. Separate each server address with a space. The first server specified is the primary server. The switch sends DNS queries to the primary server first. If that query fails, the backup servers are queried.
Step 4	**ip domain-lookup**	(Optional) Enable DNS-based hostname-to-address translation on your switch. This feature is enabled by default. If your network devices require connectivity with devices in networks for which you do not control name assignment, you can dynamically assign device names that uniquely identify your devices by using the global Internet naming scheme (DNS).
Step 5	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 6	**show running-config**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

If you use the switch IP address as its hostname, the IP address is used and no DNS query occurs. If you configure a hostname that contains no periods (.), a period followed by the default domain name is appended to the hostname before the DNS query is made to map the name to an IP address. The default domain name is the value set by the **ip domain-name** global configuration command. If there is a period (.) in the hostname, the Cisco IOS software looks up the IP address without appending any default domain name to the hostname.

To remove a domain name, use the **no ip domain-name** *name* global configuration command. To remove a name server address, use the **no ip name-server** *server-address* global configuration command. To disable DNS on the switch, use the **no ip domain-lookup** global configuration command.

Displaying the DNS Configuration

To display the DNS configuration information, use the **show running-config** privileged EXEC command.

Creating a Banner

You can configure a message-of-the-day (MOTD) and a login banner. The MOTD banner displays on all connected terminals at login and is useful for sending messages that affect all network users (such as impending system shutdowns).

The login banner also displays on all connected terminals. It appears after the MOTD banner and before the login prompts.

Note For complete syntax and usage information for the commands used in this section, see the *Cisco IOS Configuration Fundamentals Command Reference, Release 12.2*.

These sections contain this configuration information:

- Default Banner Configuration, page 5-17
- Configuring a Message-of-the-Day Login Banner, page 5-18
- Configuring a Login Banner, page 5-19

Default Banner Configuration

The MOTD and login banners are not configured.

Configuring a Message-of-the-Day Login Banner

You can create a single or multiline message banner that appears on the screen when someone logs in to the switch.

Beginning in privileged EXEC mode, follow these steps to configure a MOTD login banner:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**banner motd** *c message c*	Specify the message of the day. For *c*, enter the delimiting character of your choice, for example, a pound sign (#), and press the **Return** key. The delimiting character signifies the beginning and end of the banner text. Characters after the ending delimiter are discarded. For *message*, enter a banner message up to 255 characters. You cannot use the delimiting character in the message.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To delete the MOTD banner, use the **no banner motd** global configuration command.

This example shows how to configure a MOTD banner for the switch by using the pound sign (#) symbol as the beginning and ending delimiter:

```
Switch(config)# banner motd #
This is a secure site. Only authorized users are allowed.
For access, contact technical support.
#
Switch(config)#
```

This example shows the banner that appears from the previous configuration:

```
Unix> telnet 172.2.5.4
Trying 172.2.5.4...
Connected to 172.2.5.4.
Escape character is '^]'.

This is a secure site. Only authorized users are allowed.
For access, contact technical support.

User Access Verification

Password:
```

Configuring a Login Banner

You can configure a login banner to be displayed on all connected terminals. This banner appears after the MOTD banner and before the login prompt.

Beginning in privileged EXEC mode, follow these steps to configure a login banner:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**banner login** *c message c*	Specify the login message. For *c*, enter the delimiting character of your choice, for example, a pound sign (#), and press the **Return** key. The delimiting character signifies the beginning and end of the banner text. Characters after the ending delimiter are discarded. For *message*, enter a login message up to 255 characters. You cannot use the delimiting character in the message.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To delete the login banner, use the **no banner login** global configuration command.

This example shows how to configure a login banner for the switch by using the dollar sign ($) symbol as the beginning and ending delimiter:

```
Switch(config)# banner login $
Access for authorized users only. Please enter your username and password.
$
Switch(config)#
```

Managing the MAC Address Table

The MAC address table contains address information that the switch uses to forward traffic between ports. All MAC addresses in the address table are associated with one or more ports. The address table includes these types of addresses:

- Dynamic address: a source MAC address that the switch learns and then ages when it is not in use.
- Static address: a manually entered unicast address that does not age and that is not lost when the switch resets.

The address table lists the destination MAC address, the associated VLAN ID, and port number associated with the address and the type (static or dynamic).

Note For complete syntax and usage information for the commands used in this section, see the command reference for this release.

These sections contain this configuration information:

- Building the Address Table, page 5-20
- MAC Addresses and VLANs, page 5-20
- Default MAC Address Table Configuration, page 5-21
- Changing the Address Aging Time, page 5-21
- Removing Dynamic Address Entries, page 5-22
- Configuring MAC Address Notification Traps, page 5-22
- Adding and Removing Static Address Entries, page 5-24
- Configuring Unicast MAC Address Filtering, page 5-25
- Disabling MAC Address Learning on a VLAN, page 5-26
- Displaying Address Table Entries, page 5-28

Building the Address Table

With multiple MAC addresses supported on all ports, you can connect any port on the switch to individual workstations, repeaters, switches, routers, or other network devices. The switch provides dynamic addressing by learning the source address of packets it receives on each port and adding the address and its associated port number to the address table. As stations are added or removed from the network, the switch updates the address table, adding new dynamic addresses and aging out those that are not in use.

The aging interval is globally configured. However, the switch maintains an address table for each VLAN, and STP can accelerate the aging interval on a per-VLAN basis.

The switch sends packets between any combination of ports, based on the destination address of the received packet. Using the MAC address table, the switch forwards the packet only to the port associated with the destination address. If the destination address is on the port that sent the packet, the packet is filtered and not forwarded. The switch always uses the store-and-forward method: complete packets are stored and checked for errors before transmission.

MAC Addresses and VLANs

All addresses are associated with a VLAN. An address can exist in more than one VLAN and have different destinations in each. Unicast addresses, for example, could be forwarded to port 1 in VLAN 1 and ports 1, 9, and 10 in VLAN 5.

Each VLAN maintains its own logical address table. A known address in one VLAN is unknown in another until it is learned or statically associated with a port in the other VLAN.

When private VLANs are configured, address learning depends on the type of MAC address:

- Dynamic MAC addresses learned in one VLAN of a private VLAN are replicated in the associated VLANs. For example, a MAC address learned in a private-VLAN secondary VLAN is replicated in the primary VLAN.
- Static MAC addresses configured in a primary or secondary VLAN are not replicated in the associated VLANs. When you configure a static MAC address in a private VLAN primary or secondary VLAN, you should also configure the same static MAC address in all associated VLANs.

For more information about private VLANs, see Chapter 12, “Configuring Private VLANs.”

If the switch is running the metro IP access or metro access image, you can disable MAC address learning on a per-VLAN basis. Customers in a service provider network can tunnel a large number of MAC addresses through the network and fill up the available MAC address table space. You can control MAC address learning on a VLAN and manage the MAC address table space that is available on the switch by controlling which VLANs, and therefore which ports, can learn MAC addresses.

Before you disable MAC address learning, be sure that you are familiar with the network topology and the switch system configuration. Disabling MAC address learning on a VLAN could cause flooding in the network. See the "Disabling MAC Address Learning on a VLAN" section on page 5-26 for more information.

Default MAC Address Table Configuration

Table 5-3 shows the default MAC address table configuration.

Table 5-3 Default MAC Address Table Configuration

Feature	Default Setting
Aging time	300 seconds
Dynamic addresses	Automatically learned
Static addresses	None configured
MAC address learning on VLANs	Enabled

Changing the Address Aging Time

Dynamic addresses are source MAC addresses that the switch learns and then ages when they are not in use. You can change the aging time setting for all VLANs or for a specified VLAN.

Setting too short an aging time can cause addresses to be prematurely removed from the table. Then when the switch receives a packet for an unknown destination, it floods the packet to all ports in the same VLAN as the receiving port. This unnecessary flooding can impact performance. Setting too long an aging time can cause the address table to be filled with unused addresses, which prevents new addresses from being learned. Flooding results, which can impact switch performance.

Beginning in privileged EXEC mode, follow these steps to configure the dynamic address table aging time:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mac address-table aging-time** [**0** \| *10-1000000*] [**vlan** *vlan-id*]	Set the length of time that a dynamic entry remains in the MAC address table after the entry is used or updated. The range is 10 to 1000000 seconds. The default is 300. You can also enter 0, which disables aging. Static address entries are never aged or removed from the table. For *vlan-id*, valid IDs are 1 to 4094. Do not enter leading zeros.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show mac address-table aging-time**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default value, use the **no mac address-table aging-time** global configuration command.

Removing Dynamic Address Entries

To remove all dynamic entries, use the **clear mac address-table dynamic** command in privileged EXEC mode. You can also remove a specific MAC address (**clear mac address-table dynamic address** *mac-address*), remove all addresses on the specified physical port or port channel (**clear mac address-table dynamic interface** *interface-id*), or remove all addresses on a specified VLAN (**clear mac address-table dynamic vlan** *vlan-id*).

To verify that dynamic entries have been removed, use the **show mac address-table dynamic** privileged EXEC command.

Configuring MAC Address Notification Traps

MAC address notification enables you to track users on a network by storing the MAC address activity on the switch. Whenever the switch learns or removes a MAC address, an SNMP notification can be generated and sent to the NMS. If you have many users coming and going from the network, you can set a trap interval time to bundle the notification traps and reduce network traffic. The MAC notification history table stores the MAC address activity for each hardware port for which the trap is enabled. MAC address notifications are generated for dynamic and secure MAC addresses; events are not generated for self addresses, multicast addresses, or other static addresses.

Beginning in privileged EXEC mode, follow these steps to configure the switch to send MAC address notification traps to an NMS host:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**snmp-server host** *host-addr* {**traps** \| **informs**} {**version** {**1** \| **2c** \| **3**}} *community-string notification-type*	Specify the recipient of the trap message. • For *host-addr*, specify the name or address of the NMS. • Specify **traps** (the default) to send SNMP traps to the host. Specify **informs** to send SNMP informs to the host. • Specify the SNMP version to support. Version 1, the default, is not available with informs. • For *community-string*, specify the string to send with the notification operation. Though you can set this string by using the **snmp-server host** command, we recommend that you define this string by using the **snmp-server community** command before using the **snmp-server host** command. • For *notification-type*, use the **mac-notification** keyword.
Step 3	**snmp-server enable traps mac-notification**	Enable the switch to send MAC address traps to the NMS.
Step 4	**mac address-table notification**	Enable the MAC address notification feature.
Step 5	**mac address-table notification** [**interval** *value*] \| [**history-size** *value*]	Enter the trap interval time and the history table size. • (Optional) For **interval** *value*, specify the notification trap interval in seconds between each set of traps that are generated to the NMS. The range is 0 to 2147483647 seconds; the default is 1 second. • (Optional) For **history-size** *value*, specify the maximum number of entries in the MAC notification history table. The range is 0 to 500; the default is 1.
Step 6	**interface** *interface-id*	Enter interface configuration mode, and specify the Layer 2 interface on which to enable the SNMP MAC address notification trap.
Step 7	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled, and NNIs are enabled.
Step 8	**snmp trap mac-notification** {**added** \| **removed**}	Enable the MAC address notification trap. • Enable the MAC notification trap whenever a MAC address is **added** on this interface. • Enable the MAC notification trap whenever a MAC address is **removed** from this interface.

	Command	Purpose
Step 9	**end**	Return to privileged EXEC mode.
Step 10	**show mac address-table notification interface** **show running-config**	Verify your entries.
Step 11	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable the switch from sending MAC address notification traps, use the **no snmp-server enable traps mac-notification** global configuration command. To disable the MAC address notification traps on a specific interface, use the **no snmp trap mac-notification** {**added** | **removed**} interface configuration command. To disable the MAC address notification feature, use the **no mac address-table notification** global configuration command.

This example shows how to specify 172.20.10.10 as the NMS, enable the switch to send MAC address notification traps to the NMS, enable the MAC address notification feature, set the interval time to 60 seconds, set the history-size to 100 entries, and enable traps whenever a MAC address is added on the specified port.

```
Switch(config)# snmp-server host 172.20.10.10 traps private
Switch(config)# snmp-server enable traps mac-notification
Switch(config)# mac address-table notification
Switch(config)# mac address-table notification interval 60
Switch(config)# mac address-table notification history-size 100
Switch(config)# interface gigabitethernet0/2
Switch(config-if)# snmp trap mac-notification added
```

You can verify the previous commands by entering the **show mac address-table notification interface** and the **show mac address-table notification** privileged EXEC commands.

Adding and Removing Static Address Entries

A static address has these characteristics:

- It is manually entered in the address table and must be manually removed.
- It can be a unicast or multicast address.
- It does not age and is retained when the switch restarts.

You can add and remove static addresses and define the forwarding behavior for them. The forwarding behavior defines how a port that receives a packet forwards it to another port for transmission. Because all ports are associated with at least one VLAN, the switch acquires the VLAN ID for the address from the ports that you specify. You can specify a different list of destination ports for each source port.

A packet with a static address that arrives on a VLAN where it has not been statically entered is flooded to all ports and not learned.

You add a static address to the address table by specifying the destination MAC unicast address and the VLAN from which it is received. Packets received with this destination address are forwarded to the interface specified with the *interface-id* option.

When you configure a static MAC address in a private-VLAN primary or secondary VLAN, you should also configure the same static MAC address in all associated VLANs. Static MAC addresses configured in a private-VLAN primary or secondary VLAN are not replicated in the associated VLAN. For more information about private VLANs, see Chapter 12, “Configuring Private VLANs.”

Beginning in privileged EXEC mode, follow these steps to add a static address:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mac address-table static** *mac-addr* **vlan** *vlan-id* **interface** *interface-id*	Add a static address to the MAC address table. • For *mac-addr*, specify the destination MAC unicast address to add to the address table. Packets with this destination address received in the specified VLAN are forwarded to the specified interface. • For *vlan-id*, specify the VLAN for which the packet with the specified MAC address is received. Valid VLAN IDs are 1 to 4094; do not enter leading zeros. • For *interface-id*, specify the interface to which the received packet is forwarded. Valid interfaces include physical ports or port channels. For static multicast addresses, you can enter multiple interface IDs. For static unicast addresses, you can enter only one interface at a time, but you can enter the command multiple times with the same MAC address and VLAN ID.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show mac address-table static**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove static entries from the address table, use the **no mac address-table static** *mac-addr* **vlan** *vlan-id* [**interface** *interface-id*] global configuration command.

This example shows how to add the static address c2f3.220a.12f4 to the MAC address table. When a packet is received in VLAN 4 with this MAC address as its destination address, the packet is forwarded to the specified port:

```
Switch(config)# mac address-table static c2f3.220a.12f4 vlan 4 interface
gigabitethernet0/1
```

Configuring Unicast MAC Address Filtering

When unicast MAC address filtering is enabled, the switch drops packets with specific source or destination MAC addresses. This feature is disabled by default and only supports unicast static addresses.

Follow these guidelines when using this feature:

- Multicast MAC addresses, broadcast MAC addresses, and router MAC addresses are not supported. If you specify one of these addresses when entering the **mac address-table static** *mac-addr* **vlan** *vlan-id* **drop** global configuration command, one of these messages appears:

  ```
  % Only unicast addresses can be configured to be dropped

  % CPU destined address cannot be configured as drop address
  ```

- Packets that are forwarded to the CPU are also not supported.

- If you add a unicast MAC address as a static address and configure unicast MAC address filtering, the switch either adds the MAC address as a static address or drops packets with that MAC address, depending on which command was entered last. The second command that you entered overrides the first command.

 For example, if you enter the **mac address-table static** *mac-addr* **vlan** *vlan-id* **interface** *interface-id* global configuration command followed by the **mac address-table static** *mac-addr* **vlan** *vlan-id* **drop** command, the switch drops packets with the specified MAC address as a source or destination.

 If you enter the **mac address-table static** *mac-addr* **vlan** *vlan-id* **drop** global configuration command followed by the **mac address-table static** *mac-addr* **vlan** *vlan-id* **interface** *interface-id* command, the switch adds the MAC address as a static address.

You enable unicast MAC address filtering and configure the switch to drop packets with a specific address by specifying the source or destination unicast MAC address and the VLAN from which it is received.

Beginning in privileged EXEC mode, follow these steps to configure the switch to drop a source or destination unicast static address:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mac address-table static** *mac-addr* **vlan** *vlan-id* **drop**	Enable unicast MAC address filtering and configure the switch to drop a packet with the specified source or destination unicast static address. • For *mac-addr*, specify a source or destination unicast MAC address. Packets with this MAC address are dropped. • For *vlan-id*, specify the VLAN for which the packet with the specified MAC address is received. Valid VLAN IDs are 1 to 4094.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show mac address-table static**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable unicast MAC address filtering, use the **no mac address-table static** *mac-addr* **vlan** *vlan-id* global configuration command.

This example shows how to enable unicast MAC address filtering and to configure the switch to drop packets that have a source or destination address of c2f3.220a.12f4. When a packet is received in VLAN 4 with this MAC address as its source or destination, the packet is dropped:

```
Switch(config)# mac address-table static c2f3.220a.12f4 vlan 4 drop
```

Disabling MAC Address Learning on a VLAN

By default, MAC address learning is enabled on all VLANs on the switch. Controlling MAC address learning on a VLAN allows you to manage the MAC address table space that is available on the switch by controlling which VLANs, and therefore which ports, can learn MAC addresses. Before you disable MAC address learning be sure that you are familiar with the network topology and the switch system configuration. Disabling MAC address learning on a VLAN could cause flooding in the network.

Follow these guidelines when disabling MAC address learning on a VLAN:

- Disabling MAC address learning on a VLAN is supported only if the switch is running the metro IP access or metro access image.
- Use caution when considering disabling MAC address learning on a VLAN with a switch virtual interface (SVI) configured. If you disable MAC address learning on an SVI, the switch floods all IP packets in the Layer 2 domain.
- We recommend that you disable MAC address learning only in VLANs with two ports. If you disable MAC address learning on a VLAN with more than two ports, every packet entering the switch is flooded in that VLAN domain.
- You cannot disable MAC address learning on a VLAN that is used internally by the switch. If the VLAN ID that you enter is an internal VLAN, the switch generates an error message and rejects the command. To view internal VLANs in use by the switch, enter the **show vlan internal usage** privileged EXEC command.
- If you disable MAC address learning on a VLAN configured as a private-VLAN primary VLAN, MAC addresses are still learned on the secondary VLAN that belongs to the private VLAN and are then replicated on the primary VLAN. If you disable MAC address learning on the secondary VLAN, but not the primary VLAN of a private VLAN, MAC address learning occurs on the primary VLAN and is replicated on the secondary VLAN.
- You cannot disable MAC address learning on an RSPAN VLAN. The configuration is not allowed.
- If you disable MAC address learning on a VLAN that includes a secure port, MAC address learning is not disabled on the secure port. If you disable port security, the configured MAC address learning state is active.

Beginning in privileged EXEC mode, follow these steps to disable MAC address learning on a VLAN:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**no mac address-table learning vlan** *vlan-id*	Disable MAC address learning on the specified VLAN. Valid VLAN IDs are 1 to 4094.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show mac address-table learning** [**vlan** *vlan-id*]	Verify the configuration.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To reenable MAC address learning on a VLAN, use the **default mac address-table learning vlan** *vlan-id* global configuration command. You can also reenable MAC address learning on a VLAN by entering the **mac address-table learning vlan** *vlan-id* global configuration command. Using the second command causes the configuration to appear in the **show running-config** privileged EXEC command display. Using the first (**default**) command implies a return to a default condition and therefore does not appear in the output from the **show running-config** command.

This example shows how to disable MAC address learning on VLAN 200:

```
Switch(config)# no mac address-table learning vlan 200
```

You can display the MAC address learning status of all VLANs or a specified VLAN by entering the **show mac-address-table learning** [**vlan** *vlan-id*] privileged EXEC command.

Displaying Address Table Entries

You can display the MAC address table by using one or more of the privileged EXEC commands described in Table 5-4:

Table 5-4 Commands for Displaying the MAC Address Table

Command	Description
show ip igmp snooping groups	Displays the Layer 2 multicast entries for all VLANs or the specified VLAN.
show mac address-table address	Displays MAC address table information for the specified MAC address.
show mac address-table aging-time	Displays the aging time in all VLANs or the specified VLAN.
show mac address-table count	Displays the number of addresses present in all VLANs or the specified VLAN.
show mac address-table dynamic	Displays only dynamic MAC address table entries.
show mac address-table interface	Displays the MAC address table information for the specified interface.
show mac address-table learning	Displays MAC address learning status of all VLANs or the specified VLAN.
show mac address-table notification	Displays the MAC notification parameters and history table.
show mac address-table static	Displays only static MAC address table entries.
show mac address-table vlan	Displays the MAC address table information for the specified VLAN.

Managing the ARP Table

To communicate with a device (over Ethernet, for example), the software first must learn the 48-bit MAC address or the local data link address of that device. The process of learning the local data link address from an IP address is called *address resolution*.

The Address Resolution Protocol (ARP) associates a host IP address with the corresponding media or MAC addresses and the VLAN ID. Using an IP address, ARP finds the associated MAC address. When a MAC address is found, the IP-MAC address association is stored in an ARP cache for rapid retrieval. Then the IP datagram is encapsulated in a link-layer frame and sent over the network. Encapsulation of IP datagrams and ARP requests and replies on IEEE 802 networks other than Ethernet is specified by the Subnetwork Access Protocol (SNAP). By default, standard Ethernet-style ARP encapsulation (represented by the **arpa** keyword) is enabled on the IP interface.

ARP entries added manually to the table do not age and must be manually removed.

For CLI procedures, see the Cisco IOS Release 12.2 documentation on Cisco.com.



CHAPTER 6

Configuring SDM Templates

This chapter describes how to configure the Switch Database Management (SDM) templates on the Cisco ME 3400 Ethernet Access switch. SDM template configuration is supported only when the switch is running the metro IP access image.

Note For complete syntax and usage information for the commands used in this chapter, see the command reference for this release.

This chapter consists of these sections:

- Understanding the SDM Templates, page 6-1
- Configuring the Switch SDM Template, page 6-2
- Displaying the SDM Templates, page 6-4

Understanding the SDM Templates

If the switch is running the metro IP access image, you can use SDM templates to optimize system resources in the switch to support specific features, depending on how the switch is used in the network. The SDM templates allocate TCAM resources to support different features. You can use the default template to balance system resources or select the layer-2 template to support only Layer 2 features in hardware.

Note Switches running the metro base or metro access image support only the layer-2 template

- Layer-2—The layer-2 template maximizes system resources for Layer 2 functionality and does not support routing. You should use this template when the switch is being used for Layer-2 forwarding. When you select the layer-2 template on a switch running the metro IP access image, any routing is done through software, which overloads the CPU and severely degrades routing performance.
- Default—The default template gives balance to all functions: Layer 2 and Layer 3 (routing). This template is available only on switches running the metro IP access image. If you do not use the default template when routing is enabled on the switch, any routing is done through software, which overloads the CPU and severely degrades routing performance.

Table 6-1 shows the approximate number of each resource supported in each of the two templates for a switch running the metro IP access image. The values in the template are based on eight routed interfaces and approximately 1024 VLANs and represent the approximate hardware boundaries set when a template is selected. If a section of a hardware resource is full, all processing overflow is sent to the CPU, seriously impacting switch performance.

Table 6-1 Approximate Number of Feature Resources Allowed by Each Template

Resource	Layer-2	Default
Unicast MAC addresses	8 K	1 K
IPv4 IGMP groups + multicast routes (default only)	–	1 K
IP v4 IGMP groups (layer-2 only)	1 K	–
IPv4 multicast routes (layer-2 only)	0	–
IPv4 IGMP groups and multicast routes	1 K	–
IPv4 unicast routes	0	5 K
• Directly connected IPv4 hosts	–	1 K
• Indirect IPv4 routes	–	4 K
IPv4 policy-based routing ACEs[1]	0	512
IPv4 or MAC QoS[2] ACEs	512	512
IPv4 or MAC security ACEs	1 K	1 K

1. Access control entries
2. Quality of service

Configuring the Switch SDM Template

These sections contain this configuration information:

- Default SDM Template, page 6-2
- SDM Template Configuration Guidelines, page 6-2
- Setting the SDM Template, page 6-3

Default SDM Template

The default template for a switch running the metro IP access image is the default template.

The default (and only) template supported on switches running the metro base or metro access image is the layer-2 template.

SDM Template Configuration Guidelines

Follow these guidelines when selecting and configuring SDM templates:

- You must reload the switch for the configuration to take effect.
- If you are using the switch for Layer 2 features only, select the layer-2 template.

- Do not use the default template if you do not have routing enabled on your switch. The **sdm prefer default** global configuration command prevents other features from using the memory allocated to unicast routing in the routing template.
- You should use the default template when you plan to enable routing on the switch. If you do not use the default template when routing is enabled, routing is done through software, which overloads the CPU and severely degrades routing performance.

Setting the SDM Template

Beginning in privileged EXEC mode, follow these steps to use the SDM template to select a template on a switch running the metro IP access image:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**sdm prefer** {**default** \| **layer-2**}	Specify the SDM template to be used on the switch: The keywords have these meanings: • **default**—Gives balance to all functions. • **layer-2**—Supports Layer 2 functionality and does not support routing on the switch.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**reload**	Reload the operating system.

After the system reboots, you can use the **show sdm prefer** privileged EXEC command to verify the new template configuration. If you enter the **show sdm prefer** command before you enter the **reload** privileged EXEC command, the **show sdm prefer** command shows the template currently in use and the template that will become active after a reload.

This is an example of an output display when you have changed the template to the layer-2 template and have not reloaded the switch:

```
Switch# show sdm prefer
The current template is "default" template.
 The selected template optimizes the resources in
 the switch to support this level of features for
 8 routed interfaces and 1024 VLANs.

  number of unicast mac addresses:                  1K
  number of IPv4 IGMP groups + multicast routes:    1K
  number of IPv4 unicast routes:                    5K
    number of directly-connected IPv4 hosts:        1K
    number of indirect IPv4 routes:                 4K
  number of IPv4 policy based routing aces:         512
  number of IPv4/MAC qos aces:                      512
  number of IPv4/MAC security aces:                 1K

On next reload, template will be "layer-2" template.
```

To return to the default template, use the **no sdm prefer** global configuration command.

This example shows how to configure a switch with the layer-2 template.

```
Switch(config)# sdm prefer layer-2
Switch(config)# end
Switch# reload
Proceed with reload? [confirm]
```

Displaying the SDM Templates

Use the **show sdm prefer** privileged EXEC command with no parameters to display the active template. Use the **show sdm prefer** [**default** | **layer-2**] privileged EXEC command to display the resource numbers supported by the specified template.

This is an example of output from the **show sdm prefer default** command:

```
Switch# show sdm prefer default
"default" template:
 The selected template optimizes the resources in
 the switch to support this level of features for
 8 routed interfaces and 1024 VLANs.

  number of unicast mac addresses:                  1K
  number of IPv4 IGMP groups + multicast routes:    1K
  number of IPv4 unicast routes:                    5K
    number of directly-connected IPv4 hosts:        1K
    number of indirect IPv4 routes:                 4K
  number of IPv4 policy based routing aces:         512
  number of IPv4/MAC qos aces:                      512
  number of IPv4/MAC security aces:                 1K
```

This is an example of output from the **show sdm prefer layer-2** command:

```
Switch# show sdm prefer layer-2
"layer-2" template:
 The selected template optimizes the resources in
 the switch to support this level of features for
 8 routed interfaces and 1024 VLANs.

  number of unicast mac addresses:                  8K
  number of IPv4 IGMP groups:                       1K
  number of IPv4 multicast routes:                  0
  number of unicast IPv4 routes:                    0
  number of IPv4 policy based routing aces:         0
  number of IPv4/MAC qos aces:                      512
  number of IPv4/MAC security aces:                 1K
```



Configuring Switch-Based Authentication

This chapter describes how to configure switch-based authentication on the Cisco ME 3400 switch.

This chapter consists of these sections:

- Preventing Unauthorized Access to Your Switch, page 7-1
- Protecting Access to Privileged EXEC Commands, page 7-2
- Controlling Switch Access with TACACS+, page 7-10
- Controlling Switch Access with RADIUS, page 7-18
- Controlling Switch Access with Kerberos, page 7-32
- Configuring the Switch for Local Authentication and Authorization, page 7-36
- Configuring the Switch for Secure Shell, page 7-37

Preventing Unauthorized Access to Your Switch

You can prevent unauthorized users from reconfiguring your switch and viewing configuration information. Typically, you want network administrators to have access to your switch while you restrict access to users who dial from outside the network through an asynchronous port, connect from outside the network through a serial port, or connect through a terminal or workstation from within the local network.

To prevent unauthorized access into your switch, you should configure one or more of these security features:

- At a minimum, you should configure passwords and privileges at each switch port. These passwords are locally stored on the switch. When users attempt to access the switch through a port or line, they must enter the password specified for the port or line before they can access the switch. For more information, see the "Protecting Access to Privileged EXEC Commands" section on page 7-2.
- For an additional layer of security, you can also configure username and password pairs, which are locally stored on the switch. These pairs are assigned to lines or ports and authenticate each user before that user can access the switch. If you have defined privilege levels, you can also assign a specific privilege level (with associated rights and privileges) to each username and password pair. For more information, see the "Configuring Username and Password Pairs" section on page 7-7.
- If you want to use username and password pairs, but you want to store them centrally on a server instead of locally, you can store them in a database on a security server. Multiple networking devices can then use the same database to obtain user authentication (and, if necessary, authorization) information. For more information, see the "Controlling Switch Access with TACACS+" section on page 7-10.

Protecting Access to Privileged EXEC Commands

A simple way of providing terminal access control in your network is to use passwords and assign privilege levels. Password protection restricts access to a network or network device. Privilege levels define what commands users can enter after they have logged into a network device.

Note For complete syntax and usage information for the commands used in this section, see the *Cisco IOS Security Command Reference, Release 12.2*.

These sections contain this configuration information:

- Default Password and Privilege Level Configuration, page 7-2
- Setting or Changing a Static Enable Password, page 7-3
- Protecting Enable and Enable Secret Passwords with Encryption, page 7-4
- Disabling Password Recovery, page 7-5
- Setting a Telnet Password for a Terminal Line, page 7-6
- Configuring Username and Password Pairs, page 7-7
- Configuring Multiple Privilege Levels, page 7-8

Default Password and Privilege Level Configuration

Table 7-1 shows the default password and privilege level configuration.

Table 7-1 Default Password and Privilege Levels

Feature	Default Setting
Enable password and privilege level	No password is defined. The default is level 15 (privileged EXEC level). The password is not encrypted in the configuration file.
Enable secret password and privilege level	No password is defined. The default is level 15 (privileged EXEC level). The password is encrypted before it is written to the configuration file.
Line password	No password is defined.

Setting or Changing a Static Enable Password

The enable password controls access to the privileged EXEC mode. Beginning in privileged EXEC mode, follow these steps to set or change a static enable password:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**enable password** *password*	Define a new password or change an existing password for access to privileged EXEC mode. By default, no password is defined. For *password*, specify a string from 1 to 25 alphanumeric characters. The string cannot start with a number, is case sensitive, and allows spaces but ignores leading spaces. It can contain the question mark (?) character if you precede the question mark with the key combination Crtl-v when you create the password; for example, to create the password abc?123, do this: Enter **abc**. Enter **Crtl-v**. Enter **?123**. When the system prompts you to enter the enable password, you need not precede the question mark with the Ctrl-v; you can simply enter abc?123 at the password prompt.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file. The enable password is not encrypted and can be read in the switch configuration file.

To remove the password, use the **no enable password** global configuration command.

This example shows how to change the enable password to *l1u2c3k4y5*. The password is not encrypted and provides access to level 15 (traditional privileged EXEC mode access):

```
Switch(config)# enable password l1u2c3k4y5
```

Protecting Enable and Enable Secret Passwords with Encryption

To provide an additional layer of security, particularly for passwords that cross the network or that are stored on a Trivial File Transfer Protocol (TFTP) server, you can use either the **enable password** or **enable secret** global configuration commands. Both commands accomplish the same thing; that is, you can establish an encrypted password that users must enter to access privileged EXEC mode (the default) or any privilege level you specify.

We recommend that you use the **enable secret** command because it uses an improved encryption algorithm.

If you configure the **enable secret** command, it takes precedence over the **enable password** command; the two commands cannot be in effect simultaneously.

Beginning in privileged EXEC mode, follow these steps to configure encryption for enable and enable secret passwords:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**enable password** [**level** *level*] {*password* \| *encryption-type encrypted-password*}	Define a new password or change an existing password for access to privileged EXEC mode.
	or	or
	enable secret [**level** *level*] {*password* \| *encryption-type encrypted-password*}	Define a secret password, which is saved using a nonreversible encryption method. • (Optional) For *level*, the range is from 0 to 15. Level 1 is normal user EXEC mode privileges. The default level is 15 (privileged EXEC mode privileges). • For *password*, specify a string from 1 to 25 alphanumeric characters. The string cannot start with a number, is case sensitive, and allows spaces but ignores leading spaces. By default, no password is defined. • (Optional) For *encryption-type*, only type 5, a Cisco proprietary encryption algorithm, is available. If you specify an encryption type, you must provide an encrypted password—an encrypted password that you copy from another switch configuration. **Note** If you specify an encryption type and then enter a clear text password, you can not re-enter privileged EXEC mode. You cannot recover a lost encrypted password by any method.
Step 3	**service password-encryption**	(Optional) Encrypt the password when the password is defined or when the configuration is written. Encryption prevents the password from being readable in the configuration file.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

If both the enable and enable secret passwords are defined, users must enter the enable secret password.

Use the **level** keyword to define a password for a specific privilege level. After you specify the level and set a password, give the password only to users who need to have access at this level. Use the **privilege level** global configuration command to specify commands accessible at various levels. For more information, see the "Configuring Multiple Privilege Levels" section on page 7-8.

If you enable password encryption, it applies to all passwords including username passwords, authentication key passwords, the privileged command password, and console and virtual terminal line passwords.

To remove a password and level, use the **no enable password** [**level** *level*] or **no enable secret** [**level** *level*] global configuration command. To disable password encryption, use the **no service password-encryption** global configuration command.

This example shows how to configure the encrypted password *1FaD0$Xyti5Rkls3LoyxzS8* for privilege level 2:

```
Switch(config)# enable secret level 2 5 $1$FaD0$Xyti5Rkls3LoyxzS8
```

Disabling Password Recovery

By default, any end user with physical access to the switch can recover from a lost password by interrupting the boot process while the switch is powering on and then by entering a new password.

The password-recovery disable feature protects access to the switch password by disabling part of this functionality. When this feature is enabled, the end user can interrupt the boot process only by agreeing to set the system back to the default configuration. With password recovery disabled, you can still interrupt the boot process and change the password, but the configuration file (config.text) and the VLAN database file (vlan.dat) are deleted.

Note If you disable password recovery, we recommend that you keep a backup copy of the configuration file on a secure server in case the end user interrupts the boot process and sets the system back to default values. Do not keep a backup copy of the configuration file on the switch. We recommend that you also keep a backup copy of the VLAN database file on a secure server. When the switch is returned to the default system configuration, you can download the saved files to the switch by using the XMODEM protocol. For more information, see the "Recovering from a Lost or Forgotten Password" section on page 36-3.

Beginning in privileged EXEC mode, follow these steps to disable password recovery:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**no service password-recovery**	Disable password recovery. This setting is saved in an area of the flash memory that is accessible by the boot loader and the Cisco IOS image, but it is not part of the file system and is not accessible by any user.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show version**	Verify the configuration by checking the last few lines of the command output.

To re-enable password recovery, use the **service password-recovery** global configuration command.

Note Disabling password recovery will not work if you have set the switch to boot manually by using the **boot manual** global configuration command. This command produces the boot loader prompt (*switch:*) after the switch is power cycled.

Setting a Telnet Password for a Terminal Line

When you power-up your switch for the first time, an automatic setup program runs to assign IP information and to create a default configuration for continued use. The setup program also prompts you to configure your switch for Telnet access through a password. If you did not configure this password during the setup program, you can configure it now through the command-line interface (CLI).

Beginning in privileged EXEC mode, follow these steps to configure your switch for Telnet access:

	Command	Purpose
Step 1		Attach a PC or workstation with emulation software to the switch console port. The default data characteristics of the console port are 9600, 8, 1, no parity. You might need to press the Return key several times to see the command-line prompt.
Step 2	**enable password** *password*	Enter privileged EXEC mode.
Step 3	**configure terminal**	Enter global configuration mode.
Step 4	**line vty 0 15**	Configure the number of Telnet sessions (lines), and enter line configuration mode. There are 16 possible sessions on a command-capable switch. The 0 and 15 mean that you are configuring all 16 possible Telnet sessions.
Step 5	**password** *password*	Enter a Telnet password for the line or lines. For *password*, specify a string from 1 to 25 alphanumeric characters. The string cannot start with a number, is case sensitive, and allows spaces but ignores leading spaces. By default, no password is defined.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show running-config**	Verify your entries. The password is listed under the command **line vty 0 15**.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the password, use the **no password** global configuration command.

This example shows how to set the Telnet password to *let45me67in89*:

```
Switch(config)# line vty 10
Switch(config-line)# password let45me67in89
```

Configuring Username and Password Pairs

You can configure username and password pairs, which are locally stored on the switch. These pairs are assigned to lines or ports and authenticate each user before that user can access the switch. If you have defined privilege levels, you can also assign a specific privilege level (with associated rights and privileges) to each username and password pair.

Beginning in privileged EXEC mode, follow these steps to establish a username-based authentication system that requests a login username and a password:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**username** *name* [**privilege** *level*] {**password** *encryption-type password*}	Enter the username, privilege level, and password for each user. • For *name*, specify the user ID as one word. Spaces and quotation marks are not allowed. • (Optional) For *level*, specify the privilege level the user has after gaining access. The range is 0 to 15. Level 15 gives privileged EXEC mode access. Level 1 gives user EXEC mode access. • For *encryption-type*, enter 0 to specify that an unencrypted password will follow. Enter 7 to specify that a hidden password will follow. • For *password*, specify the password the user must enter to gain access to the switch. The password must be from 1 to 25 characters, can contain embedded spaces, and must be the last option specified in the **username** command.
Step 3	**line console 0** or **line vty 0 15**	Enter line configuration mode, and configure the console port (line 0) or the VTY lines (line 0 to 15).
Step 4	**login local**	Enable local password checking at login time. Authentication is based on the username specified in Step 2.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable username authentication for a specific user, use the **no username** *name* global configuration command. To disable password checking and allow connections without a password, use the **no login** line configuration command.

Configuring Multiple Privilege Levels

By default, the Cisco IOS software has two modes of password security: user EXEC and privileged EXEC. You can configure up to 16 hierarchical levels of commands for each mode. By configuring multiple passwords, you can allow different sets of users to have access to specified commands.

For example, if you want many users to have access to the **clear line** command, you can assign it level 2 security and distribute the level 2 password fairly widely. But if you want more restricted access to the **configure** command, you can assign it level 3 security and distribute that password to a more restricted group of users.

These sections contain this configuration information:

- Setting the Privilege Level for a Command, page 7-8
- Changing the Default Privilege Level for Lines, page 7-9
- Logging into and Exiting a Privilege Level, page 7-10

Setting the Privilege Level for a Command

Beginning in privileged EXEC mode, follow these steps to set the privilege level for a command mode:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**privilege** *mode* **level** *level command*	Set the privilege level for a command. • For *mode*, enter **configure** for global configuration mode, **exec** for EXEC mode, **interface** for interface configuration mode, or **line** for line configuration mode. • For *level*, the range is from 0 to 15. Level 1 is for normal user EXEC mode privileges. Level 15 is the level of access permitted by the **enable** password. • For *command*, specify the command to which you want to restrict access.
Step 3	**enable password level** *level password*	Specify the enable password for the privilege level. • For *level*, the range is from 0 to 15. Level 1 is for normal user EXEC mode privileges. • For *password*, specify a string from 1 to 25 alphanumeric characters. The string cannot start with a number, is case sensitive, and allows spaces but ignores leading spaces. By default, no password is defined.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config** or **show privilege**	Verify your entries. The first command shows the password and access level configuration. The second command shows the privilege level configuration.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

When you set a command to a privilege level, all commands whose syntax is a subset of that command are also set to that level. For example, if you set the **show ip traffic** command to level 15, the **show** commands and **show ip** commands are automatically set to privilege level 15 unless you set them individually to different levels.

To return to the default privilege for a given command, use the **no privilege** *mode* **level** *level command* global configuration command.

This example shows how to set the **configure** command to privilege level 14 and define *SecretPswd14* as the password users must enter to use level 14 commands:

```
Switch(config)# privilege exec level 14 configure
Switch(config)# enable password level 14 SecretPswd14
```

Changing the Default Privilege Level for Lines

Beginning in privileged EXEC mode, follow these steps to change the default privilege level for a line:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**line vty** *line*	Select the virtual terminal line on which to restrict access.
Step 3	**privilege level** *level*	Change the default privilege level for the line. For *level*, the range is from 0 to 15. Level 1 is for normal user EXEC mode privileges. Level 15 is the level of access permitted by the **enable** password.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config** or **show privilege**	Verify your entries. The first command shows the password and access level configuration. The second command shows the privilege level configuration.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Users can override the privilege level you set using the **privilege level** line configuration command by logging in to the line and enabling a different privilege level. They can lower the privilege level by using the **disable** command. If users know the password to a higher privilege level, they can use that password to enable the higher privilege level. You might specify a high level or privilege level for your console line to restrict line usage.

To return to the default line privilege level, use the **no privilege level** line configuration command.

Logging into and Exiting a Privilege Level

Beginning in privileged EXEC mode, follow these steps to log in to a specified privilege level and to exit to a specified privilege level:

	Command	Purpose
Step 1	**enable** *level*	Log in to a specified privilege level. For *level*, the range is 0 to 15.
Step 2	**disable** *level*	Exit to a specified privilege level. For *level*, the range is 0 to 15.

Controlling Switch Access with TACACS+

This section describes how to enable and configure Terminal Access Controller Access Control System Plus (TACACS+), which provides detailed accounting information and flexible administrative control over authentication and authorization processes. TACACS+ is facilitated through authentication, authorization, accounting (AAA) and can be enabled only through AAA commands.

Note For complete syntax and usage information for the commands used in this section, see the *Cisco IOS Security Command Reference, Release 12.2*.

These sections contain this configuration information:

- Understanding TACACS+, page 7-10
- TACACS+ Operation, page 7-12
- Configuring TACACS+, page 7-13
- Displaying the TACACS+ Configuration, page 7-17

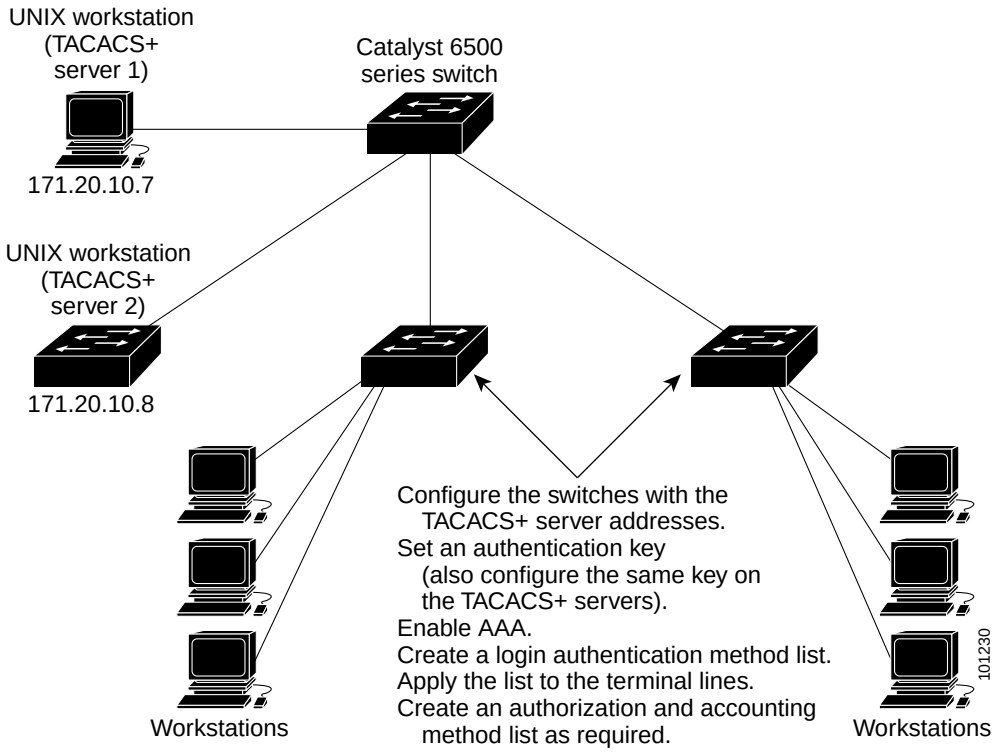
Understanding TACACS+

TACACS+ is a security application that provides centralized validation of users attempting to gain access to your switch. TACACS+ services are maintained in a database on a TACACS+ daemon typically running on a UNIX or Windows NT workstation. You should have access to and should configure a TACACS+ server before the configuring TACACS+ features on your switch.

TACACS+ provides for separate and modular authentication, authorization, and accounting facilities. TACACS+ allows for a single access control server (the TACACS+ daemon) to provide each service—authentication, authorization, and accounting—independently. Each service can be tied into its own database to take advantage of other services available on that server or on the network, depending on the capabilities of the daemon.

The goal of TACACS+ is to provide a method for managing multiple network access points from a single management service. Your switch can be a network access server along with other Cisco routers and access servers. A network access server provides connections to a single user, to a network or subnetwork, and to interconnected networks as shown in Figure 7-1.

Figure 7-1 Typical TACACS+ Network Configuration



TACACS+, administered through the AAA security services, can provide these services:

- Authentication—Provides complete control of authentication through login and password dialog, challenge and response, and messaging support.

 The authentication facility can conduct a dialog with the user (for example, after a username and password are provided, to challenge a user with several questions, such as home address, mother's maiden name, service type, and social security number). The TACACS+ authentication service can also send messages to user screens. For example, a message could notify users that their passwords must be changed because of the company's password aging policy.

- Authorization—Provides fine-grained control over user capabilities for the duration of the user's session, including but not limited to setting autocommands, access control, session duration, or protocol support. You can also enforce restrictions on what commands a user can execute with the TACACS+ authorization feature.

- Accounting—Collects and sends information used for billing, auditing, and reporting to the TACACS+ daemon. Network managers can use the accounting facility to track user activity for a security audit or to provide information for user billing. Accounting records include user identities, start and stop times, executed commands (such as PPP), number of packets, and number of bytes.

The TACACS+ protocol provides authentication between the switch and the TACACS+ daemon, and it ensures confidentiality because all protocol exchanges between the switch and the TACACS+ daemon are encrypted.

You need a system running the TACACS+ daemon software to use TACACS+ on your switch.

TACACS+ Operation

When a user attempts a simple ASCII login by authenticating to a switch using TACACS+, this process occurs:

1. When the connection is established, the switch contacts the TACACS+ daemon to obtain a username prompt to show to the user. The user enters a username, and the switch then contacts the TACACS+ daemon to obtain a password prompt. The switch displays the password prompt to the user, the user enters a password, and the password is then sent to the TACACS+ daemon.

 TACACS+ allows a dialog between the daemon and the user until the daemon receives enough information to authenticate the user. The daemon prompts for a username and password combination, but can include other items, such as the user's mother's maiden name.

2. The switch eventually receives one of these responses from the TACACS+ daemon:

 - ACCEPT—The user is authenticated and service can begin. If the switch is configured to require authorization, authorization begins at this time.
 - REJECT—The user is not authenticated. The user can be denied access or is prompted to retry the login sequence, depending on the TACACS+ daemon.
 - ERROR—An error occurred at some time during authentication with the daemon or in the network connection between the daemon and the switch. If an ERROR response is received, the switch typically tries to use an alternative method for authenticating the user.
 - CONTINUE—The user is prompted for additional authentication information.

 After authentication, the user undergoes an additional authorization phase if authorization has been enabled on the switch. Users must first successfully complete TACACS+ authentication before proceeding to TACACS+ authorization.

3. If TACACS+ authorization is required, the TACACS+ daemon is again contacted, and it returns an ACCEPT or REJECT authorization response. If an ACCEPT response is returned, the response contains data in the form of attributes that direct the EXEC or NETWORK session for that user and the services that the user can access:

 - Telnet, Secure Shell (SSH), rlogin, or privileged EXEC services
 - Connection parameters, including the host or client IP address, access list, and user timeouts

Configuring TACACS+

This section describes how to configure your switch to support TACACS+. At a minimum, you must identify the host or hosts maintaining the TACACS+ daemon and define the method lists for TACACS+ authentication. You can optionally define method lists for TACACS+ authorization and accounting. A method list defines the sequence and methods to be used to authenticate, to authorize, or to keep accounts on a user. You can use method lists to designate one or more security protocols to be used, thus ensuring a backup system if the initial method fails. The software uses the first method listed to authenticate, to authorize, or to keep accounts on users; if that method does not respond, the software selects the next method in the list. This process continues until there is successful communication with a listed method or the method list is exhausted.

These sections contain this configuration information:

- Default TACACS+ Configuration, page 7-13
- Identifying the TACACS+ Server Host and Setting the Authentication Key, page 7-13
- Configuring TACACS+ Login Authentication, page 7-14
- Configuring TACACS+ Authorization for Privileged EXEC Access and Network Services, page 7-16
- Starting TACACS+ Accounting, page 7-17

Default TACACS+ Configuration

TACACS+ and AAA are disabled by default.

To prevent a lapse in security, you cannot configure TACACS+ through a network management application. When enabled, TACACS+ can authenticate users accessing the switch through the CLI.

Note Although TACACS+ configuration is performed through the CLI, the TACACS+ server authenticates HTTP connections that have been configured with a privilege level of 15.

Identifying the TACACS+ Server Host and Setting the Authentication Key

You can configure the switch to use a single server or AAA server groups to group existing server hosts for authentication. You can group servers to select a subset of the configured server hosts and use them for a particular service. The server group is used with a global server-host list and contains the list of IP addresses of the selected server hosts.

Beginning in privileged EXEC mode, follow these steps to identify the IP host or host maintaining TACACS+ server and optionally set the encryption key:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**tacacs-server host** *hostname* [**port** *integer*] [**timeout** *integer*] [**key** *string*]	Identify the IP host or hosts maintaining a TACACS+ server. Enter this command multiple times to create a list of preferred hosts. The software searches for hosts in the order in which you specify them. • For *hostname*, specify the name or IP address of the host. • (Optional) For **port** *integer*, specify a server port number. The default is port 49. The range is 1 to 65535. • (Optional) For **timeout** *integer*, specify a time in seconds the switch waits for a response from the daemon before it times out and declares an error. The default is 5 seconds. The range is 1 to 1000 seconds. • (Optional) For **key** *string*, specify the encryption key for encrypting and decrypting all traffic between the switch and the TACACS+ daemon. You must configure the same key on the TACACS+ daemon for encryption to be successful.
Step 3	**aaa new-model**	Enable AAA.
Step 4	**aaa group server tacacs+** *group-name*	(Optional) Define the AAA server-group with a group name. This command puts the switch in a server group subconfiguration mode.
Step 5	**server** *ip-address*	(Optional) Associate a particular TACACS+ server with the defined server group. Repeat this step for each TACACS+ server in the AAA server group. Each server in the group must be previously defined in Step 2.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show tacacs**	Verify your entries.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the specified TACACS+ server name or address, use the **no tacacs-server host** *hostname* global configuration command. To remove a server group from the configuration list, use the **no aaa group server tacacs+** *group-name* global configuration command. To remove the IP address of a TACACS+ server, use the **no server ip-address** server group subconfiguration command.

Configuring TACACS+ Login Authentication

To configure AAA authentication, you define a named list of authentication methods and then apply that list to various ports. The method list defines the types of authentication to be performed and the sequence in which they are performed; it must be applied to a specific port before any of the defined authentication methods are performed. The only exception is the default method list (which, by coincidence, is named *default*). The default method list is automatically applied to all ports except those that have a named method list explicitly defined. A defined method list overrides the default method list.

A method list describes the sequence and authentication methods to be queried to authenticate a user. You can designate one or more security protocols to be used for authentication, thus ensuring a backup system for authentication in case the initial method fails. The software uses the first method listed to

authenticate users; if that method fails to respond, the software selects the next authentication method in the method list. This process continues until there is successful communication with a listed authentication method or until all defined methods are exhausted. If authentication fails at any point in this cycle—meaning that the security server or local username database responds by denying the user access—the authentication process stops, and no other authentication methods are attempted.

Beginning in privileged EXEC mode, follow these steps to configure login authentication:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**aaa new-model**	Enable AAA.
Step 3	**aaa authentication login** {**default** \| *list-name*} *method1* [*method2...*]	Create a login authentication method list. • To create a default list that is used when a named list is *not* specified in the **login authentication** command, use the **default** keyword followed by the methods that are to be used in default situations. The default method list is automatically applied to all ports. • For *list-name*, specify a character string to name the list you are creating. • For *method1...*, specify the actual method the authentication algorithm tries. The additional methods of authentication are used only if the previous method returns an error, not if it fails. Select one of these methods: • **enable**—Use the enable password for authentication. Before you can use this authentication method, you must define an enable password by using the **enable** *password* global configuration command. • **group tacacs+**—Uses TACACS+ authentication. Before you can use this authentication method, you must configure the TACACS+ server. For more information, see the "Identifying the TACACS+ Server Host and Setting the Authentication Key" section on page 7-13. • **line**—Use the line password for authentication. Before you can use this authentication method, you must define a line password. Use the **password** *password* line configuration command. • **local**—Use the local username database for authentication. You must enter username information in the database. Use the **username** *password* global configuration command. • **local-case**—Use a case-sensitive local username database for authentication. You must enter username information in the database by using the **username** *name* **password** global configuration command. • **none**—Do not use any authentication for login.
Step 4	**line** [**console** \| **tty** \| **vty**] *line-number* [*ending-line-number*]	Enter line configuration mode, and configure the lines to which you want to apply the authentication list.

	Command	Purpose
Step 5	**login authentication** {**default** \| *list-name*}	Apply the authentication list to a line or set of lines. • If you specify **default**, use the default list created with the **aaa authentication login** command. • For *list-name*, specify the list created with the **aaa authentication login** command.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show running-config**	Verify your entries.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable AAA, use the **no aaa new-model** global configuration command. To disable AAA authentication, use the **no aaa authentication login** {**default** | *list-name*} *method1* [*method2*...] global configuration command. To either disable TACACS+ authentication for logins or to return to the default value, use the **no login authentication** {**default** | *list-name*} line configuration command.

Configuring TACACS+ Authorization for Privileged EXEC Access and Network Services

AAA authorization limits the services available to a user. When AAA authorization is enabled, the switch uses information retrieved from the user's profile, which is located either in the local user database or on the security server, to configure the user's session. The user is granted access to a requested service only if the information in the user profile allows it.

You can use the **aaa authorization** global configuration command with the **tacacs+** keyword to set parameters that restrict a user's network access to privileged EXEC mode.

The **aaa authorization exec tacacs+ local** command sets these authorization parameters:

- Use TACACS+ for privileged EXEC access authorization if authentication was performed by using TACACS+.
- Use the local database if authentication was not performed by using TACACS+.

Note Authorization is bypassed for authenticated users who log in through the CLI even if authorization has been configured.

Beginning in privileged EXEC mode, follow these steps to specify TACACS+ authorization for privileged EXEC access and network services:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**aaa authorization network tacacs+**	Configure the switch for user TACACS+ authorization for all network-related service requests.
Step 3	**aaa authorization exec tacacs+**	Configure the switch for user TACACS+ authorization if the user has privileged EXEC access. The **exec** keyword might return user profile information (such as **autocommand** information).
Step 4	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable authorization, use the **no aaa authorization** {**network** | **exec**} *method1* global configuration command.

Starting TACACS+ Accounting

The AAA accounting feature tracks the services that users are accessing and the amount of network resources that they are consuming. When AAA accounting is enabled, the switch reports user activity to the TACACS+ security server in the form of accounting records. Each accounting record contains accounting attribute-value (AV) pairs and is stored on the security server. This data can then be analyzed for network management, client billing, or auditing.

Beginning in privileged EXEC mode, follow these steps to enable TACACS+ accounting for each Cisco IOS privilege level and for network services:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**aaa accounting network start-stop tacacs+**	Enable TACACS+ accounting for all network-related service requests.
Step 3	**aaa accounting exec start-stop tacacs+**	Enable TACACS+ accounting to send a start-record accounting notice at the beginning of a privileged EXEC process and a stop-record at the end.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable accounting, use the **no aaa accounting** {**network** | **exec**} {**start-stop**} *method1...* global configuration command.

Displaying the TACACS+ Configuration

To display TACACS+ server statistics, use the **show tacacs** privileged EXEC command.

Controlling Switch Access with RADIUS

This section describes how to enable and configure the RADIUS, which provides detailed accounting information and flexible administrative control over authentication and authorization processes. RADIUS is facilitated through AAA and can be enabled only through AAA commands.

Note For complete syntax and usage information for the commands used in this section, see the *Cisco IOS Security Command Reference, Release 12.2*.

These sections contain this configuration information:

- Understanding RADIUS, page 7-18
- RADIUS Operation, page 7-19
- Configuring RADIUS, page 7-20
- Displaying the RADIUS Configuration, page 7-31

Understanding RADIUS

RADIUS is a distributed client/server system that secures networks against unauthorized access. RADIUS clients run on supported Cisco routers and switches. Clients send authentication requests to a central RADIUS server, which contains all user authentication and network service access information. The RADIUS host is normally a multiuser system running RADIUS server software from Cisco (Cisco Secure Access Control Server Version 3.0), Livingston, Merit, Microsoft, or another software provider. For more information, see the RADIUS server documentation.

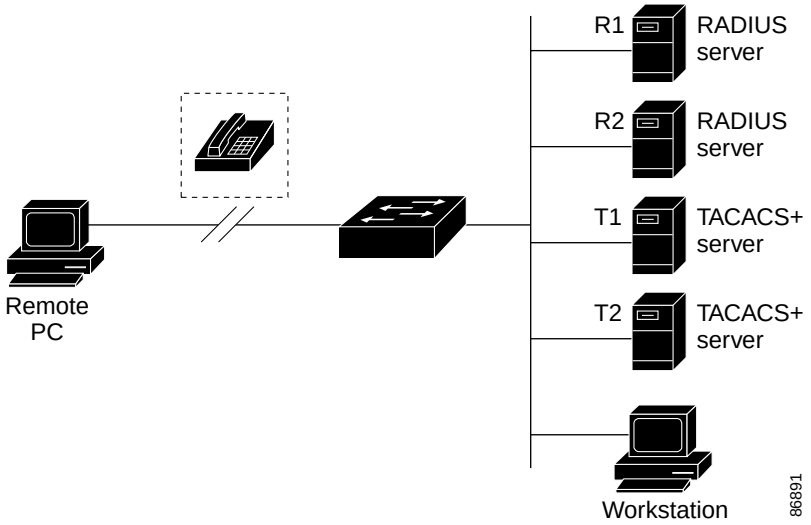
Use RADIUS in these network environments that require access security:

- Networks with multiple-vendor access servers, each supporting RADIUS. For example, access servers from several vendors use a single RADIUS server-based security database. In an IP-based network with multiple vendors' access servers, dial-in users are authenticated through a RADIUS server that has been customized to work with the Kerberos security system.
- Turnkey network security environments in which applications support the RADIUS protocol, such as in an access environment that uses a *smart card* access control system. In one case, RADIUS has been used with Enigma's security cards to validates users and to grant access to network resources.
- Networks already using RADIUS. You can add a Cisco switch containing a RADIUS client to the network. This might be the first step when you make a transition to a TACACS+ server. See Figure 7-2 on page 7-19.
- Network in which the user must only access a single service. Using RADIUS, you can control user access to a single host, to a single utility such as Telnet, or to the network through a protocol such as IEEE 802.1x. For more information about this protocol, see Chapter 8, "Configuring IEEE 802.1x Port-Based Authentication."
- Networks that require resource accounting. You can use RADIUS accounting independently of RADIUS authentication or authorization. The RADIUS accounting functions allow data to be sent at the start and end of services, showing the amount of resources (such as time, packets, bytes, and so forth) used during the session. An Internet service provider might use a freeware-based version of RADIUS access control and accounting software to meet special security and billing needs.

RADIUS is not suitable in these network security situations:

- Multiprotocol access environments. RADIUS does not support AppleTalk Remote Access (ARA), NetBIOS Frame Control Protocol (NBFCP), NetWare Asynchronous Services Interface (NASI), or X.25 PAD connections.
- Switch-to-switch or router-to-router situations. RADIUS does not provide two-way authentication. RADIUS can be used to authenticate from one device to a non-Cisco device if the non-Cisco device requires authentication.
- Networks using a variety of services. RADIUS generally binds a user to one service model.

Figure 7-2 Transitioning from RADIUS to TACACS+ Services



RADIUS Operation

When a user attempts to log in and authenticate to a switch that is access controlled by a RADIUS server, these events occur:

1. The user is prompted to enter a username and password.
2. The username and encrypted password are sent over the network to the RADIUS server.
3. The user receives one of these responses from the RADIUS server:
 a. ACCEPT—The user is authenticated.
 b. REJECT—The user is either not authenticated and is prompted to re-enter the username and password, or access is denied.
 c. CHALLENGE—A challenge requires additional data from the user.
 d. CHALLENGE PASSWORD—A response requests the user to select a new password.

The ACCEPT or REJECT response is bundled with additional data that is used for privileged EXEC or network authorization. Users must first successfully complete RADIUS authentication before proceeding to RADIUS authorization, if it is enabled. The additional data included with the ACCEPT or REJECT packets includes these items:

- Telnet, SSH, rlogin, or privileged EXEC services
- Connection parameters, including the host or client IP address, access list, and user timeouts

Configuring RADIUS

This section describes how to configure your switch to support RADIUS. At a minimum, you must identify the host or hosts that run the RADIUS server software and define the method lists for RADIUS authentication. You can optionally define method lists for RADIUS authorization and accounting.

A method list defines the sequence and methods to be used to authenticate, to authorize, or to keep accounts on a user. You can use method lists to designate one or more security protocols to be used (such as TACACS+ or local username lookup), thus ensuring a backup system if the initial method fails. The software uses the first method listed to authenticate, to authorize, or to keep accounts on users; if that method does not respond, the software selects the next method in the list. This process continues until there is successful communication with a listed method or the method list is exhausted.

You should have access to and should configure a RADIUS server before configuring RADIUS features on your switch.

These sections contain this configuration information:

- Default RADIUS Configuration, page 7-20
- Identifying the RADIUS Server Host, page 7-20 (required)
- Configuring RADIUS Login Authentication, page 7-23 (required)
- Defining AAA Server Groups, page 7-25 (optional)
- Configuring RADIUS Authorization for User Privileged Access and Network Services, page 7-27 (optional)
- Starting RADIUS Accounting, page 7-28 (optional)
- Configuring Settings for All RADIUS Servers, page 7-29 (optional)
- Configuring the Switch to Use Vendor-Specific RADIUS Attributes, page 7-29 (optional)
- Configuring the Switch for Vendor-Proprietary RADIUS Server Communication, page 7-31 (optional)

Default RADIUS Configuration

RADIUS and AAA are disabled by default.

To prevent a lapse in security, you cannot configure RADIUS through a network management application. When enabled, RADIUS can authenticate users accessing the switch through the CLI.

Identifying the RADIUS Server Host

Switch-to-RADIUS-server communication involves several components:

- Hostname or IP address
- Authentication destination port
- Accounting destination port
- Key string
- Timeout period
- Retransmission value

You identify RADIUS security servers by their hostname or IP address, hostname and specific UDP port numbers, or their IP address and specific UDP port numbers. The combination of the IP address and the UDP port number creates a unique identifier, allowing different ports to be individually defined as RADIUS hosts providing a specific AAA service. This unique identifier enables RADIUS requests to be sent to multiple UDP ports on a server at the same IP address.

If two different host entries on the same RADIUS server are configured for the same service—for example, accounting—the second host entry configured acts as a fail-over backup to the first one. Using this example, if the first host entry fails to provide accounting services, the switch tries the second host entry configured on the same device for accounting services. (The RADIUS host entries are tried in the order that they are configured.)

A RADIUS server and the switch use a shared secret text string to encrypt passwords and exchange responses. To configure RADIUS to use the AAA security commands, you must specify the host running the RADIUS server daemon and a secret text (key) string that it shares with the switch.

The timeout, retransmission, and encryption key values can be configured globally for all RADIUS servers, on a per-server basis, or in some combination of global and per-server settings. To apply these settings globally to all RADIUS servers communicating with the switch, use the three unique global configuration commands: **radius-server timeout**, **radius-server retransmit**, and **radius-server key**. To apply these values on a specific RADIUS server, use the **radius-server host** global configuration command.

You can configure the switch to use AAA server groups to group existing server hosts for authentication. For more information, see the "Defining AAA Server Groups" section on page 7-25.

Beginning in privileged EXEC mode, follow these steps to configure per-server RADIUS server communication. This procedure is required.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**aaa new-model**	Enable AAA authentication.

	Command	Purpose
Step 3	**radius-server host** {*hostname* \| *ip-address*} [**auth-port** *port-number*] [**acct-port** *port-number*] [**timeout** *seconds*] [**retransmit** *retries*] [**key** *string*]	Specify the IP address or hostname of the remote RADIUS server host. • (Optional) For **auth-port** *port-number*, specify the UDP destination port for authentication requests. • (Optional) For **acct-port** *port-number*, specify the UDP destination port for accounting requests. • (Optional) For **timeout** *seconds*, specify the time interval that the switch waits for the RADIUS server to reply before resending. The range is 1 to 1000. This setting overrides the **radius-server timeout** global configuration command setting. If no timeout is set with the **radius-server host** command, the setting of the **radius-server timeout** command is used. • (Optional) For **retransmit** *retries*, specify the number of times a RADIUS request is resent to a server if that server is not responding or responding slowly. The range is 1 to 1000. If no retransmit value is set with the **radius-server host** command, the setting of the **radius-server retransmit** global configuration command is used. • (Optional) For **key** *string*, specify the authentication and encryption key used between the switch and the RADIUS daemon running on the RADIUS server. **Note** The key is a text string that must match the encryption key used on the RADIUS server. Always configure the key as the last item in the **radius-server host** command. Leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in your key, do not enclose the key in quotation marks unless the quotation marks are part of the key. To configure the switch to recognize more than one host entry associated with a single IP address, enter this command as many times as necessary, making sure that each UDP port number is different. The switch software searches for hosts in the order in which you specify them. Set the timeout, retransmit, and encryption key values to use with the specific RADIUS host.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the specified RADIUS server, use the **no radius-server host** *hostname* | *ip-address* global configuration command.

This example shows how to configure one RADIUS server to be used for authentication and another to be used for accounting:

```
Switch(config)# radius-server host 172.29.36.49 auth-port 1612 key rad1
Switch(config)# radius-server host 172.20.36.50 acct-port 1618 key rad2
```

This example shows how to configure *host1* as the RADIUS server and to use the default ports for both authentication and accounting:

```
Switch(config)# radius-server host host1
```

Note You also need to configure some settings on the RADIUS server. These settings include the IP address of the switch and the key string to be shared by both the server and the switch. For more information, see the RADIUS server documentation.

Configuring RADIUS Login Authentication

To configure AAA authentication, you define a named list of authentication methods and then apply that list to various ports. The method list defines the types of authentication to be performed and the sequence in which they are performed; it must be applied to a specific port before any of the defined authentication methods are performed. The only exception is the default method list (which, by coincidence, is named *default*). The default method list is automatically applied to all ports except those that have a named method list explicitly defined.

A method list describes the sequence and authentication methods to be queried to authenticate a user. You can designate one or more security protocols to be used for authentication, thus ensuring a backup system for authentication in case the initial method fails. The software uses the first method listed to authenticate users; if that method fails to respond, the software selects the next authentication method in the method list. This process continues until there is successful communication with a listed authentication method or until all defined methods are exhausted. If authentication fails at any point in this cycle—meaning that the security server or local username database responds by denying the user access—the authentication process stops, and no other authentication methods are attempted.

Beginning in privileged EXEC mode, follow these steps to configure login authentication. This procedure is required.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**aaa new-model**	Enable AAA.
Step 3	**aaa authentication login** {**default** \| *list-name*} *method1* [*method2...*]	Create a login authentication method list. • To create a default list that is used when a named list is *not* specified in the **login authentication** command, use the **default** keyword followed by the methods that are to be used in default situations. The default method list is automatically applied to all ports. • For *list-name*, specify a character string to name the list you are creating. • For *method1...*, specify the actual method the authentication algorithm tries. The additional methods of authentication are used only if the previous method returns an error, not if it fails. Select one of these methods: – **enable**—Use the enable password for authentication. Before you can use this authentication method, you must define an enable password by using the **enable** *password* global configuration command. – **group radius**—Use RADIUS authentication. Before you can use this authentication method, you must configure the RADIUS server. For more information, see the "Identifying the RADIUS Server Host" section on page 7-20. – **line**—Use the line password for authentication. Before you can use this authentication method, you must define a line password. Use the **password** *password* line configuration command. – **local**—Use the local username database for authentication. You must enter username information in the database. Use the **username** *name* **password** global configuration command. – **local-case**—Use a case-sensitive local username database for authentication. You must enter username information in the database by using the **username** *password* global configuration command. – **none**—Do not use any authentication for login.
Step 4	**line** [**console** \| **tty** \| **vty**] *line-number* [*ending-line-number*]	Enter line configuration mode, and configure the lines to which you want to apply the authentication list.

	Command	Purpose
Step 5	**login authentication** {**default** \| *list-name*}	Apply the authentication list to a line or set of lines. • If you specify **default**, use the default list created with the **aaa authentication login** command. • For *list-name*, specify the list created with the **aaa authentication login** command.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show running-config**	Verify your entries.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable AAA, use the **no aaa new-model** global configuration command. To disable AAA authentication, use the **no aaa authentication login** {**default** | *list-name*} *method1* [*method2*...] global configuration command. To either disable RADIUS authentication for logins or to return to the default value, use the **no login authentication** {**default** | *list-name*} line configuration command.

Defining AAA Server Groups

You can configure the switch to use AAA server groups to group existing server hosts for authentication. You select a subset of the configured server hosts and use them for a particular service. The server group is used with a global server-host list, which lists the IP addresses of the selected server hosts.

Server groups also can include multiple host entries for the same server if each entry has a unique identifier (the combination of the IP address and UDP port number), allowing different ports to be individually defined as RADIUS hosts providing a specific AAA service. If you configure two different host entries on the same RADIUS server for the same service, (for example, accounting), the second configured host entry acts as a fail-over backup to the first one.

You use the **server** group server configuration command to associate a particular server with a defined group server. You can either identify the server by its IP address or identify multiple host instances or entries by using the optional **auth-port** and **acct-port** keywords.

Beginning in privileged EXEC mode, follow these steps to define the AAA server group and associate a particular RADIUS server with it:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**radius-server host** {*hostname* \| *ip-address*} [**auth-port** *port-number*] [**acct-port** *port-number*] [**timeout** *seconds*] [**retransmit** *retries*] [**key** *string*]	Specify the IP address or hostname of the remote RADIUS server host. • (Optional) For **auth-port** *port-number*, specify the UDP destination port for authentication requests. • (Optional) For **acct-port** *port-number*, specify the UDP destination port for accounting requests. • (Optional) For **timeout** *seconds*, specify the time interval that the switch waits for the RADIUS server to reply before resending. The range is 1 to 1000. This setting overrides the **radius-server timeout** global configuration command setting. If no timeout is set with the **radius-server host** command, the setting of the **radius-server timeout** command is used. • (Optional) For **retransmit** *retries*, specify the number of times a RADIUS request is resent to a server if that server is not responding or responding slowly. The range is 1 to 1000. If no retransmit value is set with the **radius-server host** command, the setting of the **radius-server retransmit** global configuration command is used. • (Optional) For **key** *string*, specify the authentication and encryption key used between the switch and the RADIUS daemon running on the RADIUS server. **Note** The key is a text string that must match the encryption key used on the RADIUS server. Always configure the key as the last item in the **radius-server host** command. Leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in your key, do not enclose the key in quotation marks unless the quotation marks are part of the key. To configure the switch to recognize more than one host entry associated with a single IP address, enter this command as many times as necessary, making sure that each UDP port number is different. The switch software searches for hosts in the order in which you specify them. Set the timeout, retransmit, and encryption key values to use with the specific RADIUS host.
Step 3	**aaa new-model**	Enable AAA.
Step 4	**aaa group server radius** *group-name*	Define the AAA server-group with a group name. This command puts the switch in a server group configuration mode.
Step 5	**server** *ip-address*	Associate a particular RADIUS server with the defined server group. Repeat this step for each RADIUS server in the AAA server group. Each server in the group must be previously defined in Step 2.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show running-config**	Verify your entries.

	Command	Purpose
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.
Step 9		Enable RADIUS login authentication. See the "Configuring RADIUS Login Authentication" section on page 7-23.

To remove the specified RADIUS server, use the **no radius-server host** *hostname* | *ip-address* global configuration command. To remove a server group from the configuration list, use the **no aaa group server radius** *group-name* global configuration command. To remove the IP address of a RADIUS server, use the **no server** *ip-address* server group configuration command.

In this example, the switch is configured to recognize two different RADIUS group servers (*group1* and *group2*). Group1 has two different host entries on the same RADIUS server configured for the same services. The second host entry acts as a fail-over backup to the first entry.

```
Switch(config)# radius-server host 172.20.0.1 auth-port 1000 acct-port 1001
Switch(config)# radius-server host 172.10.0.1 auth-port 1645 acct-port 1646
Switch(config)# aaa new-model
Switch(config)# aaa group server radius group1
Switch(config-sg-radius)# server 172.20.0.1 auth-port 1000 acct-port 1001
Switch(config-sg-radius)# exit
Switch(config)# aaa group server radius group2
Switch(config-sg-radius)# server 172.20.0.1 auth-port 2000 acct-port 2001
Switch(config-sg-radius)# exit
```

Configuring RADIUS Authorization for User Privileged Access and Network Services

AAA authorization limits the services available to a user. When AAA authorization is enabled, the switch uses information retrieved from the user's profile, which is in the local user database or on the security server, to configure the user's session. The user is granted access to a requested service only if the information in the user profile allows it.

You can use the **aaa authorization** global configuration command with the **radius** keyword to set parameters that restrict a user's network access to privileged EXEC mode.

The **aaa authorization exec radius local** command sets these authorization parameters:

- Use RADIUS for privileged EXEC access authorization if authentication was performed by using RADIUS.
- Use the local database if authentication was not performed by using RADIUS.

Note Authorization is bypassed for authenticated users who log in through the CLI even if authorization has been configured.

Beginning in privileged EXEC mode, follow these steps to specify RADIUS authorization for privileged EXEC access and network services:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**aaa authorization network radius**	Configure the switch for user RADIUS authorization for all network-related service requests.
Step 3	**aaa authorization exec radius**	Configure the switch for user RADIUS authorization if the user has privileged EXEC access. The **exec** keyword might return user profile information (such as **autocommand** information).
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable authorization, use the **no aaa authorization** {**network** | **exec**} *method1* global configuration command.

Starting RADIUS Accounting

The AAA accounting feature tracks the services that users are accessing and the amount of network resources that they are consuming. When AAA accounting is enabled, the switch reports user activity to the RADIUS security server in the form of accounting records. Each accounting record contains accounting attribute-value (AV) pairs and is stored on the security server. This data can then be analyzed for network management, client billing, or auditing.

Beginning in privileged EXEC mode, follow these steps to enable RADIUS accounting for each Cisco IOS privilege level and for network services:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**aaa accounting network start-stop radius**	Enable RADIUS accounting for all network-related service requests.
Step 3	**aaa accounting exec start-stop radius**	Enable RADIUS accounting to send a start-record accounting notice at the beginning of a privileged EXEC process and a stop-record at the end.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable accounting, use the **no aaa accounting** {**network** | **exec**} {**start-stop**} *method1...* global configuration command.

Configuring Settings for All RADIUS Servers

Beginning in privileged EXEC mode, follow these steps to configure global communication settings between the switch and all RADIUS servers:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**radius-server key** *string*	Specify the shared secret text string used between the switch and all RADIUS servers. **Note** The key is a text string that must match the encryption key used on the RADIUS server. Leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in your key, do not enclose the key in quotation marks unless the quotation marks are part of the key.
Step 3	**radius-server retransmit** *retries*	Specify the number of times the switch sends each RADIUS request to the server before giving up. The default is 3; the range 1 to 1000.
Step 4	**radius-server timeout** *seconds*	Specify the number of seconds a switch waits for a reply to a RADIUS request before resending the request. The default is 5 seconds; the range is 1 to 1000.
Step 5	**radius-server deadtime** *minutes*	Specify the number of minutes a RADIUS server, which is not responding to authentication requests, to be skipped, thus avoiding the wait for the request to timeout before trying the next configured server. The default is 0; the range is 1 to 1440 minutes.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show running-config**	Verify your settings.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting for the retransmit, timeout, and deadtime, use the **no** forms of these commands.

Configuring the Switch to Use Vendor-Specific RADIUS Attributes

The Internet Engineering Task Force (IETF) draft standard specifies a method for communicating vendor-specific information between the switch and the RADIUS server by using the vendor-specific attribute (attribute 26). Vendor-specific attributes (VSAs) allow vendors to support their own extended attributes not suitable for general use. The Cisco RADIUS implementation supports one vendor-specific option by using the format recommended in the specification. Cisco's vendor-ID is 9, and the supported option has vendor-type 1, which is named *cisco-avpair*. The value is a string with this format:

```
protocol : attribute sep value *
```

Protocol is a value of the Cisco protocol attribute for a particular type of authorization. *Attribute* and *value* are an appropriate attribute-value (AV) pair defined in the Cisco TACACS+ specification, and *sep* is = for mandatory attributes and is * for optional attributes. The full set of features available for TACACS+ authorization can then be used for RADIUS.

For example, this AV pair activates Cisco's *multiple named ip address pools* feature during IP authorization (during PPP IPCP address assignment):

```
cisco-avpair= "ip:addr-pool=first"
```

This example shows how to provide a user logging in from a switch with immediate access to privileged EXEC commands:

```
cisco-avpair= "shell:priv-lvl=15"
```

This example shows how to specify an authorized VLAN in the RADIUS server database:

```
cisco-avpair= "tunnel-type(#64)=VLAN(13)"
cisco-avpair= "tunnel-medium-type(#65)=802 media(6)"
cisco-avpair= "tunnel-private-group-ID(#81)=vlanid"
```

This example shows how to apply an input ACL in ASCII format to an interface for the duration of this connection:

```
cisco-avpair= "ip:inacl#1=deny ip 10.10.10.10 0.0.255.255 20.20.20.20 255.255.0.0"
cisco-avpair= "ip:inacl#2=deny ip 10.10.10.10 0.0.255.255 any"
cisco-avpair= "mac:inacl#3=deny any any decnet-iv"
```

This example shows how to apply an output ACL in ASCII format to an interface for the duration of this connection:

```
cisco-avpair= "ip:outacl#2=deny ip 10.10.10.10 0.0.255.255 any"
```

Other vendors have their own unique vendor-IDs, options, and associated VSAs. For more information about vendor-IDs and VSAs, see RFC 2138, "Remote Authentication Dial-In User Service (RADIUS)."

Beginning in privileged EXEC mode, follow these steps to configure the switch to recognize and use VSAs:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**radius-server vsa send** [**accounting** \| **authentication**]	Enable the switch to recognize and use VSAs as defined by RADIUS IETF attribute 26. • (Optional) Use the **accounting** keyword to limit the set of recognized vendor-specific attributes to only accounting attributes. • (Optional) Use the **authentication** keyword to limit the set of recognized vendor-specific attributes to only authentication attributes. If you enter this command without keywords, both accounting and authentication vendor-specific attributes are used.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your settings.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

For a complete list of RADIUS attributes or more information about vendor-specific attribute 26, see the "RADIUS Attributes" appendix in the *Cisco IOS Security Configuration Guide, Release 12.2*.

Configuring the Switch for Vendor-Proprietary RADIUS Server Communication

Although an IETF draft standard for RADIUS specifies a method for communicating vendor-proprietary information between the switch and the RADIUS server, some vendors have extended the RADIUS attribute set in a unique way. Cisco IOS software supports a subset of vendor-proprietary RADIUS attributes.

As mentioned earlier, to configure RADIUS (whether vendor-proprietary or IETF draft-compliant), you must specify the host running the RADIUS server daemon and the secret text string it shares with the switch. You specify the RADIUS host and secret text string by using the **radius-server** global configuration commands.

Beginning in privileged EXEC mode, follow these steps to specify a vendor-proprietary RADIUS server host and a shared secret text string:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**radius-server host** {*hostname* \| *ip-address*} **non-standard**	Specify the IP address or hostname of the remote RADIUS server host and identify that it is using a vendor-proprietary implementation of RADIUS.
Step 3	**radius-server key** *string*	Specify the shared secret text string used between the switch and the vendor-proprietary RADIUS server. The switch and the RADIUS server use this text string to encrypt passwords and exchange responses. **Note** The key is a text string that must match the encryption key used on the RADIUS server. Leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in your key, do not enclose the key in quotation marks unless the quotation marks are part of the key.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your settings.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To delete the vendor-proprietary RADIUS host, use the **no radius-server host** {*hostname* | *ip-address*} **non-standard** global configuration command. To disable the key, use the **no radius-server key** global configuration command.

This example shows how to specify a vendor-proprietary RADIUS host and to use a secret key of *rad124* between the switch and the server:

```
Switch(config)# radius-server host 172.20.30.15 nonstandard
Switch(config)# radius-server key rad124
```

Displaying the RADIUS Configuration

To display the RADIUS configuration, use the **show running-config** privileged EXEC command.

Controlling Switch Access with Kerberos

This section describes how to enable and configure the Kerberos security system, which authenticates requests for network resources by using a trusted third party. To use this feature, the cryptographic (that is, supports encryption) version of the switch software must be installed on your switch. You must obtain authorization to use this feature and to download the cryptographic software files from Cisco.com. For more information, see the release notes for this release.

These sections contain this information:

- Understanding Kerberos, page 7-32
- Kerberos Operation, page 7-34
- Configuring Kerberos, page 7-36

For Kerberos configuration examples, see the "Kerberos Configuration Examples" section in the "Security Server Protocols" chapter of the *Cisco IOS Security Configuration Guide, Release 12.2*, at this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fsecur_c/fsecsp/

Note For complete syntax and usage information for the commands used in this section, see the "Kerberos Commands" section in the "Security Server Protocols" chapter of the *Cisco IOS Security Command Reference, Release 12.2*, at this URL: http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fsecur_c/fsecsp/index.htm.

Note In the Kerberos configuration examples and in the *Cisco IOS Security Command Reference, Release 12.2*, the trusted third party can be a Cisco ME switch that supports Kerberos, that is configured as a network security server, and that can authenticate users by using the Kerberos protocol.

Understanding Kerberos

Kerberos is a secret-key network authentication protocol, which was developed at the Massachusetts Institute of Technology (MIT). It uses the Data Encryption Standard (DES) cryptographic algorithm for encryption and authentication and authenticates requests for network resources. Kerberos uses the concept of a trusted third party to perform secure verification of users and services. This trusted third party is called the *key distribution center* (KDC).

Kerberos verifies that users are who they claim to be and the network services that they use are what the services claim to be. To do this, a KDC or trusted Kerberos server issues tickets to users. These tickets, which have a limited lifespan, are stored in user credential caches. The Kerberos server uses the tickets instead of usernames and passwords to authenticate users and network services.

Note A Kerberos server can be a Cisco ME switch that is configured as a network security server and that can authenticate users by using the Kerberos protocol.

The Kerberos credential scheme uses a process called *single logon*. This process authenticates a user once and then allows secure authentication (without encrypting another password) wherever that user credential is accepted.

This software release supports Kerberos 5, which allows organizations that are already using Kerberos 5 to use the same Kerberos authentication database on the KDC that they are already using on their other network hosts (such as UNIX servers and PCs).

In this software release, Kerberos supports these network services:

- Telnet
- rlogin
- rsh (Remote Shell Protocol)

Table 7-2 lists the common Kerberos-related terms and definitions:

Table 7-2 Kerberos Terms

Term	Definition
Authentication	A process by which a user or service identifies itself to another service. For example, a client can authenticate to a switch or a switch can authenticate to another switch.
Authorization	A means by which the switch identifies what privileges the user has in a network or on the switch and what actions the user can perform.
Credential	A general term that refers to authentication tickets, such as TGTs[1] and service credentials. Kerberos credentials verify the identity of a user or service. If a network service decides to trust the Kerberos server that issued a ticket, it can be used in place of re-entering a username and password. Credentials have a default lifespan of eight hours.
Instance	An authorization level label for Kerberos principals. Most Kerberos principals are of the form *user@REALM* (for example, smith@EXAMPLE.COM). A Kerberos principal with a Kerberos instance has the form *user/instance@REALM* (for example, smith/admin@EXAMPLE.COM). The Kerberos instance can be used to specify the authorization level for the user if authentication is successful. The server of each network service might implement and enforce the authorization mappings of Kerberos instances but is not required to do so. **Note** The Kerberos principal and instance names *must* be in all lowercase characters. **Note** The Kerberos realm name *must* be in all uppercase characters.
KDC[2]	Key distribution center that consists of a Kerberos server and database program that is running on a network host.
Kerberized	A term that describes applications and services that have been modified to support the Kerberos credential infrastructure.
Kerberos realm	A domain consisting of users, hosts, and network services that are registered to a Kerberos server. The Kerberos server is trusted to verify the identity of a user or network service to another user or network service. **Note** The Kerberos realm name *must* be in all uppercase characters.
Kerberos server	A daemon that is running on a network host. Users and network services register their identity with the Kerberos server. Network services query the Kerberos server to authenticate to other network services.

Table 7-2 Kerberos Terms (continued)

Term	Definition
KEYTAB[3]	A password that a network service shares with the KDC. In Kerberos 5 and later Kerberos versions, the network service authenticates an encrypted service credential by using the KEYTAB to decrypt it. In Kerberos versions earlier than Kerberos 5, KEYTAB is referred to as SRVTAB[4].
Principal	Also known as a Kerberos identity, this is who you are or what a service is according to the Kerberos server. **Note** The Kerberos principal name *must* be in all lowercase characters.
Service credential	A credential for a network service. When issued from the KDC, this credential is encrypted with the password shared by the network service and the KDC. The password is also shared with the user TGT.
SRVTAB	A password that a network service shares with the KDC. In Kerberos 5 or later Kerberos versions, SRVTAB is referred to as KEYTAB.
TGT	Ticket granting ticket that is a credential that the KDC issues to authenticated users. When users receive a TGT, they can authenticate to network services within the Kerberos realm represented by the KDC.

1. TGT = ticket granting ticket
2. KDC = key distribution center
3. KEYTAB = key table
4. SRVTAB = server table

Kerberos Operation

A Kerberos server can be a Cisco ME switch that is configured as a network security server and that can authenticate remote users by using the Kerberos protocol. Although you can customize Kerberos in a number of ways, remote users attempting to access network services must pass through three layers of security before they can access network services.

To authenticate to network services by using a Cisco ME switch as a Kerberos server, remote users must follow these steps:

1. Authenticating to a Boundary Switch, page 7-35
2. Obtaining a TGT from a KDC, page 7-35
3. Authenticating to Network Services, page 7-35

Authenticating to a Boundary Switch

This section describes the first layer of security through which a remote user must pass. The user must first authenticate to the boundary switch. This process then occurs:

1. The user opens an un-Kerberized Telnet connection to the boundary switch.
2. The switch prompts the user for a username and password.
3. The switch requests a TGT from the KDC for this user.
4. The KDC sends an encrypted TGT that includes the user identity to the switch.
5. The switch attempts to decrypt the TGT by using the password that the user entered.
 - If the decryption is successful, the user is authenticated to the switch.
 - If the decryption is not successful, the user repeats Step 2 either by re-entering the username and password (noting if Caps Lock or Num Lock is on or off) or by entering a different username and password.

A remote user who initiates a un-Kerberized Telnet session and authenticates to a boundary switch is inside the firewall, but the user must still authenticate directly to the KDC before getting access to the network services. The user must authenticate to the KDC because the TGT that the KDC issues is stored on the switch and cannot be used for additional authentication until the user logs on to the switch.

Obtaining a TGT from a KDC

This section describes the second layer of security through which a remote user must pass. The user must now authenticate to a KDC and obtain a TGT from the KDC to access network services.

For instructions about how to authenticate to a KDC, see the "Obtaining a TGT from a KDC" section in the "Security Server Protocols" chapter of the *Cisco IOS Security Configuration Guide, Release 12.2*, at this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fsecur_c/fsecsp/scfkerb.htm#1000999.

Authenticating to Network Services

This section describes the third layer of security through which a remote user must pass. The user with a TGT must now authenticate to the network services in a Kerberos realm.

For instructions about how to authenticate to a network service, see the "Authenticating to Network Services" section in the "Security Server Protocols" chapter of the *Cisco IOS Security Configuration Guide, Release 12.2*, at this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fsecur_c/fsecsp/scfkerb.htm#1001010.

Configuring Kerberos

So that remote users can authenticate to network services, you must configure the hosts and the KDC in the Kerberos realm to communicate and mutually authenticate users and network services. To do this, you must identify them to each other. You add entries for the hosts to the Kerberos database on the KDC and add KEYTAB files generated by the KDC to all hosts in the Kerberos realm. You also create entries for the users in the KDC database.

When you add or create entries for the hosts and users, follow these guidelines:

- The Kerberos principal name *must* be in all lowercase characters.
- The Kerberos instance name *must* be in all lowercase characters.
- The Kerberos realm name *must* be in all uppercase characters.

Note A Kerberos server can be a Cisco ME switch that is configured as a network security server and that can authenticate users by using the Kerberos protocol.

To set up a Kerberos-authenticated server-client system, follow these steps:

- Configure the KDC by using Kerberos commands.
- Configure the switch to use the Kerberos protocol.

For instructions, see the "Kerberos Configuration Task List" section in the "Security Server Protocols" chapter of the *Cisco IOS Security Configuration Guide, Release 12.2*, at this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fsecur_c/fsecsp/scfkerb.htm#1001027.

Configuring the Switch for Local Authentication and Authorization

You can configure AAA to operate without a server by setting the switch to implement AAA in local mode. The switch then handles authentication and authorization. No accounting is available in this configuration.

Beginning in privileged EXEC mode, follow these steps to configure the switch for local AAA:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**aaa new-model**	Enable AAA.
Step 3	**aaa authentication login default local**	Set the login authentication to use the local username database. The **default** keyword applies the local user database authentication to all ports.
Step 4	**aaa authorization exec local**	Configure user AAA authorization, check the local database, and allow the user to run an EXEC shell.
Step 5	**aaa authorization network local**	Configure user AAA authorization for all network-related service requests.

	Command	Purpose
Step 6	**username** *name* [**privilege** *level*] {**password** *encryption-type password*}	Enter the local database, and establish a username-based authentication system. Repeat this command for each user. • For *name*, specify the user ID as one word. Spaces and quotation marks are not allowed. • (Optional) For *level*, specify the privilege level the user has after gaining access. The range is 0 to 15. Level 15 gives privileged EXEC mode access. Level 0 gives user EXEC mode access. • For *encryption-type*, enter 0 to specify that an unencrypted password follows. Enter 7 to specify that a hidden password follows. • For *password*, specify the password the user must enter to gain access to the switch. The password must be from 1 to 25 characters, can contain embedded spaces, and must be the last option specified in the **username** command.
Step 7	**end**	Return to privileged EXEC mode.
Step 8	**show running-config**	Verify your entries.
Step 9	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable AAA, use the **no aaa new-model** global configuration command. To disable authorization, use the **no aaa authorization** {**network** | **exec**} *method1* global configuration command.

Configuring the Switch for Secure Shell

This section describes how to configure the Secure Shell (SSH) feature. To use this feature, you must install the cryptographic (encrypted) software image on your switch. You must obtain authorization to use this feature and to download the cryptographic software files from Cisco.com. For more information, see the release notes for this release.

These sections contain this information:

- Understanding SSH, page 7-38
- Configuring SSH, page 7-39
- Displaying the SSH Configuration and Status, page 7-41

For SSH configuration examples, see the “SSH Configuration Examples” section in the “Configuring Secure Shell” chapter of the *Cisco IOS Security Configuration Guide, Cisco IOS Release 12.2*, at this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fsecur_c/fothersf/scfssh.htm

Note For complete syntax and usage information for the commands used in this section, see the command reference for this release and the command reference for Cisco IOS Release 12.2 at this URL: http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/index.htm.

Understanding SSH

SSH is a protocol that provides a secure, remote connection to a device. SSH provides more security for remote connections than Telnet does by providing strong encryption when a device is authenticated. This software release supports SSH Version 1 (SSHv1) and SSH Version 2 (SSHv2).

This section consists of these topics:

- SSH Servers, Integrated Clients, and Supported Versions, page 7-38
- Limitations, page 7-38

SSH Servers, Integrated Clients, and Supported Versions

The SSH feature has an SSH server and an SSH integrated client, which are applications that run on the switch. You can use an SSH client to connect to a switch running the SSH server. The SSH server works with the SSH client supported in this release and with non-Cisco SSH clients. The SSH client also works with the SSH server supported in this release and with non-Cisco SSH servers.

The switch supports an SSHv1 or an SSHv2 server.

The switch supports an SSHv1 client.

SSH supports the Data Encryption Standard (DES) encryption algorithm, the Triple DES (3DES) encryption algorithm, and password-based user authentication.

SSH also supports these user authentication methods:

- TACACS+ (for more information, see the "Controlling Switch Access with TACACS+" section on page 7-10)
- RADIUS (for more information, see the "Controlling Switch Access with RADIUS" section on page 7-18)
- Local authentication and authorization (for more information, see the "Configuring the Switch for Local Authentication and Authorization" section on page 7-36)

Note This software release does not support IP Security (IPSec).

Limitations

These limitations apply to SSH:

- The switch supports Rivest, Shamir, and Adelman (RSA) authentication.
- SSH supports only the execution-shell application.
- The SSH server and the SSH client are supported only on DES (56-bit) and 3DES (168-bit) data encryption software.
- The switch does not support the Advanced Encryption Standard (AES) symmetric encryption algorithm.

Configuring SSH

This section has this configuration information:

- Configuration Guidelines, page 7-39
- Setting Up the Switch to Run SSH, page 7-39 (required)
- Configuring the SSH Server, page 7-40 (required only if you are configuring the switch as an SSH server)

Configuration Guidelines

Follow these guidelines when configuring the switch as an SSH server or SSH client:

- An RSA key pair generated by a SSHv1 server can be used by an SSHv2 server, and the reverse.
- If you get CLI error messages after entering the **crypto key generate rsa** global configuration command, an RSA key pair has not been generated. Reconfigure the hostname and domain, and then enter the **crypto key generate rsa** command. For more information, see the "Setting Up the Switch to Run SSH" section on page 7-39.
- When generating the RSA key pair, the message `No host name specified` might appear. If it does, you must configure a hostname by using the **hostname** global configuration command.
- When generating the RSA key pair, the message `No domain specified` might appear. If it does, you must configure an IP domain name by using the **ip domain-name** global configuration command.
- When configuring the local authentication and authorization authentication method, make sure that AAA is disabled on the console.

Setting Up the Switch to Run SSH

Follow these steps to set up your switch to run SSH:

1. Download the cryptographic software image from Cisco.com. This step is required. For more information, see the release notes for this release.
2. Configure a hostname and IP domain name for the switch. Follow this procedure only if you are configuring the switch as an SSH server.
3. Generate an RSA key pair for the switch, which automatically enables SSH. Follow this procedure only if you are configuring the switch as an SSH server.
4. Configure user authentication for local or remote access. This step is required. For more information, see the "Configuring the Switch for Local Authentication and Authorization" section on page 7-36.

Beginning in privileged EXEC mode, follow these steps to configure a hostname and an IP domain name and to generate an RSA key pair. This procedure is required if you are configuring the switch as an SSH server.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**hostname** *hostname*	Configure a hostname for your switch.
Step 3	**ip domain-name** *domain_name*	Configure a host domain for your switch.

	Command	Purpose
Step 4	**crypto key generate rsa**	Enable the SSH server for local and remote authentication on the switch and generate an RSA key pair. We recommend that a minimum modulus size of 1024 bits. When you generate RSA keys, you are prompted to enter a modulus length. A longer modulus length might be more secure, but it takes longer to generate and to use.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show ip ssh** or **show ssh**	Show the version and configuration information for your SSH server. Show the status of the SSH server on the switch.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To delete the RSA key pair, use the **crypto key zeroize rsa** global configuration command. After the RSA key pair is deleted, the SSH server is automatically disabled.

Configuring the SSH Server

Beginning in privileged EXEC mode, follow these steps to configure the SSH server:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip ssh version** [**1** \| **2**]	(Optional) Configure the switch to run SSH Version 1 or SSH Version 2. • **1**—Configure the switch to run SSH Version 1. • **2**—Configure the switch to run SSH Version 2. If you do not enter this command or do not specify a keyword, the SSH server selects the latest SSH version supported by the SSH client. For example, if the SSH client supports SSHv1 and SSHv2, the SSH server selects SSHv2.
Step 3	**ip ssh** {**timeout** *seconds* \| **authentication-retries** *number*}	Configure the SSH control parameters: • Specify the time-out value in seconds; the default is 120 seconds. The range is 0 to 120 seconds. This parameter applies to the SSH negotiation phase. After the connection is established, the switch uses the default time-out values of the CLI-based sessions. By default, up to five simultaneous, encrypted SSH connections for multiple CLI-based sessions over the network are available (session 0 to session 4). After the execution shell starts, the CLI-based session time-out value returns to the default of 10 minutes. • Specify the number of times that a client can re-authenticate to the server. The default is 3; the range is 0 to 5. Repeat this step when configuring both parameters.
Step 4	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 5	**show ip ssh**	Show the version and configuration information for your SSH server.
	or	
	show ssh	Show the status of the SSH server connections on the switch.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default SSH control parameters, use the **no ip ssh** {**timeout** | **authentication-retries**} global configuration command.

Displaying the SSH Configuration and Status

To display the SSH server configuration and status, use one or more of the privileged EXEC commands in Table 7-3:

Table 7-3 Commands for Displaying the SSH Server Configuration and Status

Command	Purpose
show ip ssh	Shows the version and configuration information for the SSH server.
show ssh	Shows the status of the SSH server.

For more information about these commands, see the "Secure Shell Commands" section in the "Other Security Features" chapter of the *Cisco IOS Security Command Reference, Cisco IOS Release 12.2*, at this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fsecur_r/fothercr/srfssh.htm.



Configuring IEEE 802.1x Port-Based Authentication

This chapter describes how to configure IEEE 802.1x port-based authentication on the Cisco ME 3400 Ethernet Access switch. As LANs extend to hotels, airports, and corporate lobbies and create insecure environments, 802.1x prevents unauthorized devices (clients) from gaining access to the network.

Note For complete syntax and usage information for the commands used in this chapter, see the command reference for this release.

This chapter consists of these sections:

- Understanding IEEE 802.1x Port-Based Authentication, page 8-1
- Configuring IEEE 802.1x Authentication, page 8-9
- Displaying IEEE 802.1x Statistics and Status, page 8-19

Understanding IEEE 802.1x Port-Based Authentication

The IEEE 802.1x standard defines a client-server-based access control and authentication protocol that prevents unauthorized clients from connecting to a LAN through publicly accessible ports unless they are properly authenticated. The authentication server authenticates each client connected to a switch port before making available any services offered by the switch or the LAN.

Until the client is authenticated, IEEE 802.1x access control allows only Extensible Authentication Protocol over LAN (EAPOL), Cisco Discovery Protocol (CDP), and Spanning Tree Protocol (STP) traffic through the port to which the client is connected. After authentication is successful, normal traffic can pass through the port.

Note CDP and STP are only supported on network node interfaces (NNIs). User network nodes (UNIs) do not support CDP or STP.

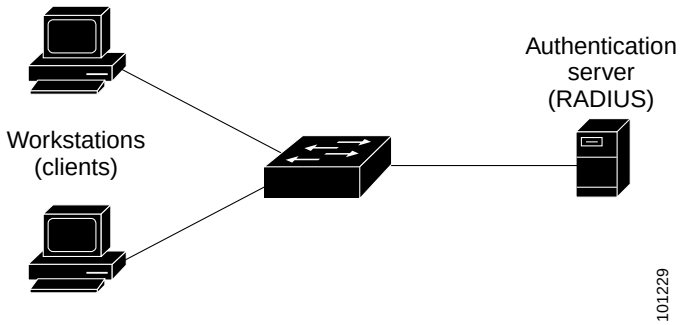
These sections describe IEEE 802.1x port-based authentication:

- Device Roles, page 8-2
- Authentication Initiation and Message Exchange, page 8-3
- Ports in Authorized and Unauthorized States, page 8-4
- IEEE 802.1x Accounting, page 8-5
- IEEE 802.1x Accounting Attribute-Value Pairs, page 8-5
- IEEE 802.1x Host Mode, page 8-6
- Using IEEE 802.1x with Port Security, page 8-7
- Using IEEE 802.1x with VLAN Assignment, page 8-8
- Using IEEE 802.1x with VLAN Assignment, page 8-8

Device Roles

With IEEE 802.1x port-based authentication, the devices in the network have specific roles as shown in Figure 8-1.

Figure 8-1 IEEE 802.1x Device Roles



- *Client*—the device (workstation) that requests access to the LAN and switch services and responds to requests from the switch. The workstation must be running IEEE 802.1x-compliant client software such as that offered in the Microsoft Windows XP operating system. (The client is the *supplicant* in the IEEE 802.1x specification.)

 Note To resolve Windows XP network connectivity and IEEE 802.1x authentication issues, read the Microsoft Knowledge Base article at this URL: http://support.microsoft.com/support/kb/articles/Q303/5/97.ASP

- *Authentication server*—performs the actual authentication of the client. The authentication server validates the identity of the client and notifies the switch whether or not the client is authorized to access the LAN and switch services. Because the switch acts as the proxy, the authentication service is transparent to the client. In this release, the RADIUS security system with Extensible Authentication Protocol (EAP) extensions is the only supported authentication server. It is available in Cisco Secure Access Control Server Version 3.0 or later. RADIUS operates in a client/server model in which secure authentication information is exchanged between the RADIUS server and one or more RADIUS clients.

- *Switch* (edge switch or wireless access point)—controls the physical access to the network based on the authentication status of the client. The switch acts as an intermediary (proxy) between the client and the authentication server, requesting identity information from the client, verifying that information with the authentication server, and relaying a response to the client. The switch includes the RADIUS client, which is responsible for encapsulating and decapsulating the EAP frames and interacting with the authentication server.

 When the switch receives EAPOL frames and relays them to the authentication server, the Ethernet header is stripped, and the remaining EAP frame is re-encapsulated in the RADIUS format. The EAP frames are not modified during encapsulation, and the authentication server must support EAP within the native frame format. When the switch receives frames from the authentication server, the server's frame header is removed, leaving the EAP frame, which is then encapsulated for Ethernet and sent to the client.

Authentication Initiation and Message Exchange

The switch or the client can initiate authentication. If you enable authentication on a port by using the **dot1x port-control auto** interface configuration command, the switch initiates authentication when the link state changes from down to up or periodically as long as the port remains up and unauthenticated. The switch sends an EAP-request/identity frame to the client to request its identity. Upon receipt of the frame, the client responds with an EAP-response/identity frame.

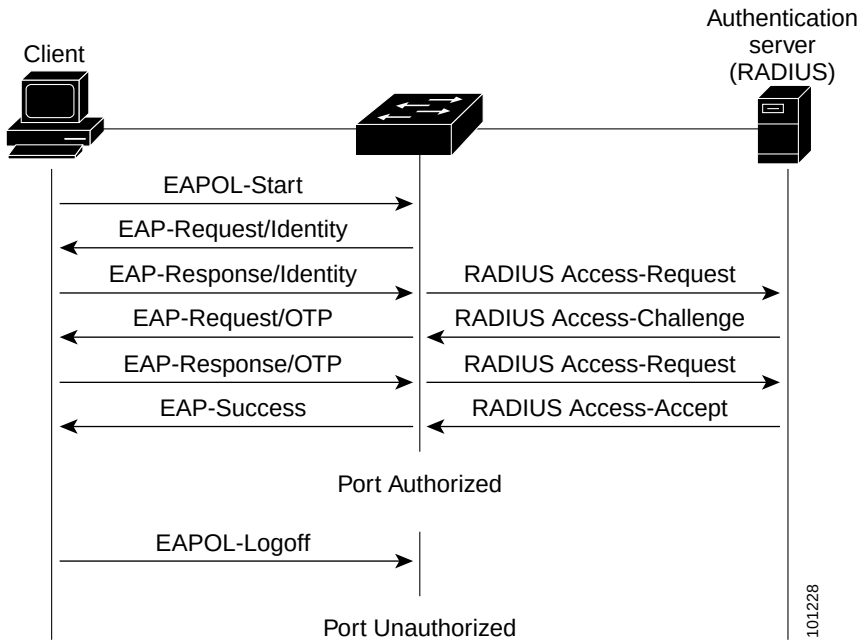
However, if during bootup, the client does not receive an EAP-request/identity frame from the switch, the client can initiate authentication by sending an EAPOL-start frame, which prompts the switch to request the client's identity.

Note If IEEE 802.1x is not enabled or supported on the network access device, any EAPOL frames from the client are dropped. If the client does not receive an EAP-request/identity frame after three attempts to start authentication, the client sends frames as if the port is in the authorized state. A port in the authorized state effectively means that the client has been successfully authenticated. For more information, see the "Ports in Authorized and Unauthorized States" section on page 8-4.

When the client supplies its identity, the switch begins its role as the intermediary, passing EAP frames between the client and the authentication server until authentication succeeds or fails. If the authentication succeeds, the switch port becomes authorized. For more information, see the "Ports in Authorized and Unauthorized States" section on page 8-4.

The specific exchange of EAP frames depends on the authentication method being used. Figure 8-2 shows a message exchange initiated by the client using the One-Time-Password (OTP) authentication method with a RADIUS server.

Figure 8-2 Message Exchange



Ports in Authorized and Unauthorized States

Depending on the switch port state, the switch can grant a client access to the network. The port starts in the *unauthorized* state. While in this state, the port disallows all incoming and outgoing traffic except for IEEE 802.1x, CDP, and STP packets. When a client is successfully authenticated, the port changes to the *authorized* state, allowing all traffic for the client to flow normally.

If a client that does not support IEEE 802.1x connects to an unauthorized IEEE 802.1x port, the switch requests the client's identity. In this situation, the client does not respond to the request, the port remains in the unauthorized state, and the client is not granted access to the network.

In contrast, when an IEEE 802.1x-enabled client connects to a port that is not running the IEEE 802.1x standard, the client initiates the authentication process by sending the EAPOL-start frame. When no response is received, the client sends the request for a fixed number of times. Because no response is received, the client begins sending frames as if the port is in the authorized state.

You control the port authorization state by using the **dot1x port-control** interface configuration command and these keywords:

- **force-authorized**—disables IEEE 802.1x authentication and causes the port to change to the authorized state without any authentication exchange required. The port sends and receives normal traffic without IEEE 802.1x-based authentication of the client. This is the default setting.
- **force-unauthorized**—causes the port to remain in the unauthorized state, ignoring all attempts by the client to authenticate. The switch cannot provide authentication services to the client through the port.
- **auto**—enables IEEE 802.1x authentication and causes the port to begin in the unauthorized state, allowing only EAPOL frames to be sent and received through the port. The authentication process begins when the link state of the port changes from down to up or when an EAPOL-start frame is received. The switch requests the identity of the client and begins relaying authentication messages between the client and the authentication server. Each client attempting to access the network is uniquely identified by the switch by using the client MAC address.

If the client is successfully authenticated (receives an Accept frame from the authentication server), the port state changes to authorized, and all frames from the authenticated client are allowed through the port. If the authentication fails, the port remains in the unauthorized state, but authentication can be retried. If the authentication server cannot be reached, the switch can resend the request. If no response is received from the server after the specified number of attempts, authentication fails, and network access is not granted.

When a client logs off, it sends an EAPOL-logoff message, causing the switch port to change to the unauthorized state.

If the link state of a port changes from up to down, or if an EAPOL-logoff frame is received, the port returns to the unauthorized state.

IEEE 802.1x Accounting

The IEEE 802.1x standard defines how users are authorized and authenticated for network access but does not keep track of network usage. IEEE 802.1x accounting is disabled by default. You can enable IEEE 802.1x accounting to monitor this activity on IEEE 802.1x-enabled ports:

- User successfully authenticates.
- User logs off.
- Link-down occurs.
- Re-authentication successfully occurs.
- Re-authentication fails.

The switch does not log IEEE 802.1x accounting information. Instead, it sends this information to the RADIUS server, which must be configured to log accounting messages.

IEEE 802.1x Accounting Attribute-Value Pairs

The information sent to the RADIUS server is represented in the form of Attribute-Value (AV) pairs. These AV pairs provide data for different applications. (For example, a billing application might require information that is contained within the Acct-Input-Octets or the Acct-Output-Octets of a packet.)

You do not need to configure AV pairs. These are automatically sent by a switch that is configured for IEEE 802.1x accounting. Table 8-1 lists the AV pairs that might be sent by the switch:

Table 8-1 Accounting AV Pairs

Attribute number	AV pair name
Attribute[1]	User-Name
Attribute[4]	NAS-IP-Address
Attribute[5]	NAS-Port
Attribute[6]	NAS-Port-Type
Attribute[8]	Framed-IP-Address
Attribute[25]	Class
Attribute[30]	Called-Station-ID
Attribute[31]	Calling-Station-ID

Table 8-1 Accounting AV Pairs (continued)

Attribute number	AV pair name
Attribute[40]	Acct-Status-Type
Attribute[41]	Acct-Delay-Time
Attribute[42]	Acct-Input-Octets
Attribute[43]	Acct-Output-Octets
Attribute[44]	Acct-Session-ID
Attribute[45]	Acct-Authentic
Attribute[46]	Acct-Session-Time
Attribute[49]	Acct-Terminate-Cause

You can view the AV pairs that are being sent by the switch by enabling the **debug radius accounting** or **debug aaa accounting** privileged EXEC commands. For more information about these commands, see the *Cisco IOS Debug Command Reference, Release 12.2* at this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122sup/122debug/

See RFC 3580, "IEEE 802.1X Remote Authentication Dial In User Service (RADIUS) Usage Guidelines," for more information about AV pairs.

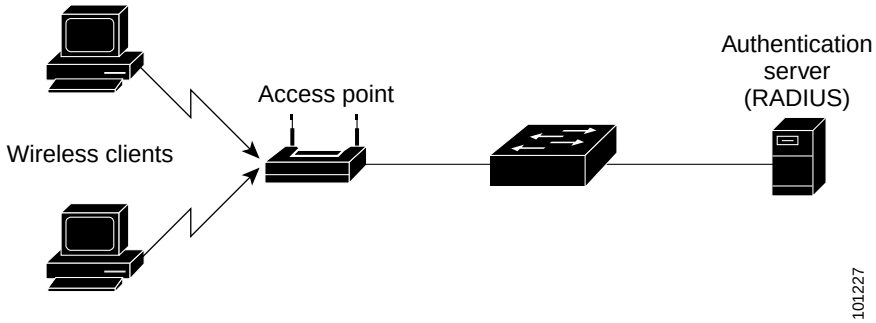
IEEE 802.1x Host Mode

You can configure an IEEE 802.1x port for single-host or for multiple-hosts mode. In single-host mode (see Figure 8-1 on page 8-2), only one client can be connected to the IEEE 802.1x-enabled switch port. The switch detects the client by sending an EAPOL frame when the port link state changes to the up state. If a client leaves or is replaced with another client, the switch changes the port link state to down, and the port returns to the unauthorized state.

In multiple-hosts mode, you can attach multiple hosts to a single IEEE 802.1x-enabled port. Figure 8-3 on page 8-7 shows IEEE 802.1x port-based authentication in a wireless LAN. In this mode, only one of the attached clients must be authorized for all clients to be granted network access. If the port becomes unauthorized (re-authentication fails or an EAPOL-logoff message is received), the switch denies network access to all of the attached clients. In this topology, the wireless access point is responsible for authenticating the clients attached to it, and it also acts as a client to the switch.

With the multiple-hosts mode enabled, you can use IEEE 802.1x to authenticate the port and port security to manage network access for all MAC addresses, including that of the client.

Figure 8-3 Multiple Host Mode Example



Using IEEE 802.1x with Port Security

You can configure an IEEE 802.1x port with port security in either single-host or multiple-hosts mode. (You also must configure port security on the port by using the **switchport port-security** interface configuration command.) When you enable port security and IEEE 802.1x on a port, IEEE 802.1x authenticates the port, and port security manages network access for all MAC addresses, including that of the client. You can then limit the number or group of clients that can access the network through an IEEE 802.1x port.

These are some examples of the interaction between IEEE 802.1x and port security on the switch:

- When a client is authenticated, and the port security table is not full, the client MAC address is added to the port security list of secure hosts. The port then proceeds to come up normally.

 When a client is authenticated and manually configured for port security, it is guaranteed an entry in the secure host table (unless port security static aging has been enabled).

 A security violation occurs if the client is authenticated, but the port security table is full. This can happen if the maximum number of secure hosts has been statically configured or if the client ages out of the secure host table. If the client address is aged, its place in the secure host table can be taken by another host.

 If the security violation is caused by the first authenticated host, the port becomes error-disabled and immediately shuts down.

 The port security violation modes determine the action for security violations. For more information, see the "Security Violations" section on page 21-9.
- When you manually remove an IEEE 802.1x client address from the port security table by using the **no switchport port-security mac-address** *mac-address* interface configuration command, you should re-authenticate the IEEE 802.1x client by using the **dot1x re-authenticate interface** *interface-id* privileged EXEC command.
- When an IEEE 802.1x client logs off, the port changes to an unauthenticated state, and all dynamic entries in the secure host table are cleared, including the entry for the client. Normal authentication then takes place.
- If the port is administratively shut down, the port becomes unauthenticated, and all dynamic entries are removed from the secure host table.

For more information about enabling port security on your switch, see the "Configuring Port Security" section on page 21-8.

Using IEEE 802.1x with VLAN Assignment

The RADIUS server sends the VLAN assignment to configure the switch port. The RADIUS server database maintains the username-to-VLAN mappings, assigning the VLAN based on the username of the client connected to the switch port. You can use this feature to limit network access for certain users.

When configured on the switch and the RADIUS server, IEEE 802.1x with VLAN assignment has these characteristics:

- If no VLAN is supplied by the RADIUS server or if IEEE 802.1x authorization is disabled, the port is configured in its access VLAN after successful authentication.
- If IEEE 802.1x authorization is enabled but the VLAN information from the RADIUS server is not valid, the port returns to the unauthorized state and remains in the configured access VLAN. This prevents ports from appearing unexpectedly in an inappropriate VLAN because of a configuration error.

 Configuration errors could include specifying a VLAN for a routed port, a malformed VLAN ID, or a nonexistent or internal (routed port) VLAN ID.
- If IEEE 802.1x authorization is enabled and all information from the RADIUS server is valid, the port is placed in the specified VLAN after authentication.
- If the multiple-hosts mode is enabled on an IEEE 802.1x port, all hosts are placed in the same VLAN (specified by the RADIUS server) as the first authenticated host.
- If IEEE 802.1x and port security are enabled on a port, the port is placed in the RADIUS server-assigned VLAN.
- If IEEE 802.1x is disabled on the port, it is returned to the configured access VLAN.

When the port is in the force authorized, the force unauthorized, the unauthorized, or the shutdown state, it is put into the configured access VLAN.

If an IEEE 802.1x port is authenticated and put in the RADIUS server-assigned VLAN, any change to the port access VLAN configuration does not take effect.

The IEEE 802.1x with VLAN assignment feature is not supported on trunk ports or with dynamic-access port assignment through a VLAN Membership Policy Server (VMPS).

To configure VLAN assignment you need to perform these tasks:

- Enable AAA authorization by using the **network** keyword to allow interface configuration from the RADIUS server.
- Enable IEEE 802.1x. (The VLAN assignment feature is automatically enabled when you configure IEEE 802.1x on an access port).
- Assign vendor-specific tunnel attributes in the RADIUS server. The RADIUS server must return these attributes to the switch:
 - [64] Tunnel-Type = VLAN
 - [65] Tunnel-Medium-Type = 802
 - [81] Tunnel-Private-Group-ID = VLAN name or VLAN ID

 Attribute[64] must contain the value *VLAN* (type 13). Attribute[65] must contain the value *802* (type 6). Attribute[81] specifies the *VLAN name* or *VLAN ID* assigned to the IEEE 802.1x-authenticated user.

For examples of tunnel attributes, see the "Configuring the Switch to Use Vendor-Specific RADIUS Attributes" section on page 7-29.

Configuring IEEE 802.1x Authentication

These sections contain this configuration information:

- Default IEEE 802.1x Configuration, page 8-9
- IEEE 802.1x Configuration Guidelines, page 8-10
- Configuring IEEE 802.1x Authentication, page 8-11 (required)
- Configuring the Switch-to-RADIUS-Server Communication, page 8-12 (required)
- Configuring Periodic Re-Authentication, page 8-13 (optional)
- Manually Re-Authenticating a Client Connected to a Port, page 8-14 (optional)
- Changing the Quiet Period, page 8-14 (optional)
- Changing the Switch-to-Client Retransmission Time, page 8-15 (optional)
- Setting the Switch-to-Client Frame-Retransmission Number, page 8-15 (optional)
- Setting the Re-Authentication Number, page 8-16 (optional)
- Configuring the Host Mode, page 8-17 (optional)
- Resetting the IEEE 802.1x Configuration to the Default Values, page 8-17 (optional)
- Configuring IEEE 802.1x Accounting, page 8-18 (optional)

Default IEEE 802.1x Configuration

Table 8-2 shows the default IEEE 802.1x configuration.

Table 8-2 Default IEEE 802.1x Configuration

Feature	Default Setting
AAA	Disabled.
RADIUS server • IP address • UDP authentication port • Key	 • None specified. • 1812. • None specified.
Switch IEEE 802.1x enable state	Disabled.
Per-port IEEE 802.1x enable state	Disabled (force-authorized). The port sends and receives normal traffic without IEEE 802.1x-based authentication of the client.
Periodic re-authentication	Disabled.
Number of seconds between re-authentication attempts	3600 seconds.
Re-authentication number	2 times (number of times that the switch restarts the authentication process before the port changes to the unauthorized state).

Table 8-2 Default IEEE 802.1x Configuration (continued)

Feature	Default Setting
Quiet period	60 seconds (number of seconds that the switch remains in the quiet state following a failed authentication exchange with the client).
Retransmission time	30 seconds (number of seconds that the switch should wait for a response to an EAP request/identity frame from the client before resending the request).
Maximum retransmission number	2 times (number of times that the switch will send an EAP-request/identity frame before restarting the authentication process).
Host mode	Single-host mode.
Client timeout period	30 seconds (when relaying a request from the authentication server to the client, the amount of time the switch waits for a response before resending the request to the client.)
Authentication server timeout period	30 seconds (when relaying a response from the client to the authentication server, the amount of time the switch waits for a reply before resending the response to the server. This setting is not configurable.)

IEEE 802.1x Configuration Guidelines

These are the IEEE 802.1x authentication configuration guidelines:

- When IEEE 802.1x is enabled, ports are authenticated before any other Layer 2 or Layer 3 features are enabled.
- The IEEE 802.1x protocol is supported on Layer 2 static-access ports and Layer 3 routed ports, but it is not supported on these port types:
 - Trunk port—If you try to enable IEEE 802.1x on a trunk port, an error message appears, and IEEE 802.1x is not enabled. If you try to change the mode of an IEEE 802.1x-enabled port to trunk, an error message appears, and the port mode is not changed.
 - Dynamic-access ports—If you try to enable IEEE 802.1x on a dynamic-access (VLAN Query Protocol [VQP]) port, an error message appears, and IEEE 802.1x is not enabled. If you try to change an IEEE 802.1x-enabled port to dynamic VLAN assignment, an error message appears, and the VLAN configuration is not changed.
 - EtherChannel port—Do not configure a port that is an active or a not-yet-active member of an EtherChannel as an IEEE 802.1x port. If you try to enable IEEE 802.1x on an EtherChannel port, an error message appears, and IEEE 802.1x is not enabled.
 - Switched Port Analyzer (SPAN) and Remote SPAN (RSPAN) destination ports—You can enable IEEE 802.1x on a port that is a SPAN or RSPAN destination port. However, IEEE 802.1x is disabled until the port is removed as a SPAN or RSPAN destination port. You can enable IEEE 802.1x on a SPAN or RSPAN source port.
- You can configure any VLAN except an RSPAN VLAN or a private VLAN.
- The IEEE 802.1x with VLAN assignment feature is not supported on private-VLAN ports, trunk ports, or ports with dynamic-access port assignment through a VMPS.

- You can configure IEEE 802.1x on a private-VLAN port, but do not configure IEEE 802.1x with port security on private-VLAN ports.
- Before globally enabling IEEE 802.1x on a switch by entering the **dot1x system-auth-control** global configuration command, remove the EtherChannel configuration from the interfaces on which IEEE 802.1x and EtherChannel are configured.

Configuring IEEE 802.1x Authentication

To configure IEEE 802.1x port-based authentication, you must enable authentication, authorization, and accounting (AAA) and specify the authentication method list. A method list describes the sequence and authentication method to be queried to authenticate a user.

To allow VLAN assignment, you must enable AAA authorization to configure the switch for all network-related service requests.

This is the IEEE 802.1x AAA process:

Step 1 A user connects to a port on the switch.

Step 2 Authentication is performed.

Step 3 VLAN assignment is enabled, as appropriate, based on the RADIUS server configuration.

Step 4 The switch sends a start message to an accounting server.

Step 5 Re-authentication is performed, as necessary.

Step 6 The switch sends an interim accounting update to the accounting server that is based on the result of re-authentication.

Step 7 The user disconnects from the port.

Step 8 The switch sends a stop message to the accounting server.

Beginning in privileged EXEC mode, follow these steps to configure IEEE 802.1x port-based authentication:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**aaa new-model**	Enable AAA.
Step 3	**aaa authentication dot1x** {**default**} *method1*	Create an IEEE 802.1x authentication method list. To create a default list that is used when a named list is *not* specified in the **authentication** command, use the **default** keyword followed by the method that is to be used in default situations. The default method list is automatically applied to all ports. For *method1*, enter the **group radius** keywords to use the list of all RADIUS servers for authentication. **Note** Though other keywords are visible in the command-line help string, only the **group radius** keywords are supported.
Step 4	**dot1x system-auth-control**	Enable IEEE 802.1x authentication globally on the switch.

	Command	Purpose
Step 5	**aaa authorization network** {**default**} **group radius**	(Optional) Configure the switch for user RADIUS authorization for all network-related service requests, such as VLAN assignment.
Step 6	**interface** *interface-id*	Specify the port connected to the client that is to be enabled for IEEE 802.1x authentication, and enter interface configuration mode.
Step 7	**dot1x port-control auto**	Enable IEEE 802.1x authentication on the port. For feature interaction information, see the "IEEE 802.1x Configuration Guidelines" section on page 8-10.
Step 8	**end**	Return to privileged EXEC mode.
Step 9	**show dot1x**	Verify your entries.
Step 10	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Configuring the Switch-to-RADIUS-Server Communication

RADIUS security servers are identified by their hostname or IP address, hostname and specific UDP port numbers, or IP address and specific UDP port numbers. The combination of the IP address and UDP port number creates a unique identifier, which enables RADIUS requests to be sent to multiple UDP ports on a server at the same IP address. If two different host entries on the same RADIUS server are configured for the same service—for example, authentication—the second host entry configured acts as the fail-over backup to the first one. The RADIUS host entries are tried in the order that they were configured.

Beginning in privileged EXEC mode, follow these steps to configure the RADIUS server parameters on the switch. This procedure is required.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**radius-server host** {*hostname* \| *ip-address*} **auth-port** *port-number* **key** *string*	Configure the RADIUS server parameters. For *hostname* \| *ip-address*, specify the hostname or IP address of the remote RADIUS server. For **auth-port** *port-number*, specify the UDP destination port for authentication requests. The default is 1812. The range is 0 to 65536. For **key** *string*, specify the authentication and encryption key used between the switch and the RADIUS daemon running on the RADIUS server. The key is a text string that must match the encryption key used on the RADIUS server. **Note** Always configure the key as the last item in the **radius-server host** command syntax because leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in the key, do not enclose the key in quotation marks unless the quotation marks are part of the key. This key must match the encryption used on the RADIUS daemon. If you want to use multiple RADIUS servers, re-enter this command.
Step 3	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To delete the specified RADIUS server, use the **no radius-server host** {*hostname* | *ip-address*} global configuration command.

This example shows how to specify the server with IP address 172.20.39.46 as the RADIUS server, to use port 1612 as the authorization port, and to set the encryption key to *rad123*, matching the key on the RADIUS server:

```
Switch(config)# radius-server host 172.l20.39.46 auth-port 1612 key rad123
```

You can globally configure the timeout, retransmission, and encryption key values for all RADIUS servers by using the **radius-server host** global configuration command. If you want to configure these options on a per-server basis, use the **radius-server timeout**, **radius-server retransmit**, and the **radius-server key** global configuration commands. For more information, see the "Configuring Settings for All RADIUS Servers" section on page 7-29.

You also need to configure some settings on the RADIUS server. These settings include the IP address of the switch and the key string to be shared by both the server and the switch. For more information, see the RADIUS server documentation.

Configuring Periodic Re-Authentication

You can enable periodic IEEE 802.1x client re-authentication and specify how often it occurs. If you do not specify a time period before enabling re-authentication, the number of seconds between re-authentication attempts is 3600.

Beginning in privileged EXEC mode, follow these steps to enable periodic re-authentication of the client and to configure the number of seconds between re-authentication attempts. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the port to be configured, and enter interface configuration mode.
Step 3	**dot1x reauthentication**	Enable periodic re-authentication of the client, which is disabled by default.
Step 4	**dot1x timeout reauth-period** *seconds*	Set the number of seconds between re-authentication attempts. The range is 1 to 65535; the default is 3600 seconds. This command affects the behavior of the switch only if periodic re-authentication is enabled.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show dot1x interface** *interface-id*	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable periodic re-authentication, use the **no dot1x reauthentication** interface configuration command. To return to the default number of seconds between re-authentication attempts, use the **no dot1x timeout reauth-period** interface configuration command.

This example shows how to enable periodic re-authentication and set the number of seconds between re-authentication attempts to 4000:

```
Switch(config-if)# dot1x reauthentication
Switch(config-if)# dot1x timeout reauth-period 4000
```

Manually Re-Authenticating a Client Connected to a Port

You can manually re-authenticate the client connected to a specific port at any time by entering the **dot1x re-authenticate interface** *interface-id* privileged EXEC command. This step is optional. If you want to enable or disable periodic re-authentication, see the "Configuring Periodic Re-Authentication" section on page 8-13.

This example shows how to manually re-authenticate the client connected to a port:

```
Switch# dot1x re-authenticate interface gigabitethernet0/1
```

Changing the Quiet Period

When the switch cannot authenticate the client, the switch remains idle for a set period of time and then tries again. The **dot1x timeout quiet-period** interface configuration command controls the idle period. A failed authentication of the client might occur because the client provided an invalid password. You can provide a faster response time to the user by entering a smaller number than the default.

Beginning in privileged EXEC mode, follow these steps to change the quiet period. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the port to be configured, and enter interface configuration mode.
Step 3	**dot1x timeout quiet-period** *seconds*	Set the number of seconds that the switch remains in the quiet state following a failed authentication exchange with the client. The range is 1 to 65535 seconds; the default is 60.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show dot1x interface** *interface-id*	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default quiet time, use the **no dot1x timeout quiet-period** interface configuration command.

This example shows how to set the quiet time on the switch to 30 seconds:

```
Switch(config-if)# dot1x timeout quiet-period 30
```

Changing the Switch-to-Client Retransmission Time

The client responds to the EAP-request/identity frame from the switch with an EAP-response/identity frame. If the switch does not receive this response, it waits a set period of time (known as the retransmission time) and then resends the frame.

Note You should change the default value of this command only to adjust for unusual circumstances such as unreliable links or specific behavioral problems with certain clients and authentication servers.

Beginning in privileged EXEC mode, follow these steps to change the amount of time that the switch waits for client notification. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the port to be configured, and enter interface configuration mode.
Step 3	**dot1x timeout tx-period** *seconds*	Set the number of seconds that the switch waits for a response to an EAP-request/identity frame from the client before resending the request. The range is 15 to 65535 seconds; the default is 30.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show dot1xinterface** *interface-id*	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default retransmission time, use the **no dot1x timeout tx-period** interface configuration command.

This example shows how to set 60 as the number of seconds that the switch waits for a response to an EAP-request/identity frame from the client before resending the request:

```
Switch(config-if)# dot1x timeout tx-period 60
```

Setting the Switch-to-Client Frame-Retransmission Number

In addition to changing the switch-to-client retransmission time, you can change the number of times that the switch sends an EAP frame (assuming no response is received) to the client before restarting the authentication process.

Note You should change the default value of this command only to adjust for unusual circumstances such as unreliable links or specific behavioral problems with certain clients and authentication servers.

Beginning in privileged EXEC mode, follow these steps to set the switch-to-client frame-retransmission number. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the port to be configured, and enter interface configuration mode.

	Command	Purpose
Step 3	**dot1x max-reauth-req** *count*	Set the number of times that the switch sends an EAP frame to the client before restarting the authentication process. The range is 1 to 10; the default is 2.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show dot1x interface** *interface-id*	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default retransmission number, use the **no dot1x max-req** interface configuration command.

This example shows how to set 5 as the number of times that the switch sends an EAP request before restarting the authentication process:

```
Switch(config-if)# dot1x max-req 5
```

Setting the Re-Authentication Number

You can also change the number of times that the switch restarts the authentication process before the port changes to the unauthorized state.

Note You should change the default value of this command only to adjust for unusual circumstances such as unreliable links or specific behavioral problems with certain clients and authentication servers.

Beginning in privileged EXEC mode, follow these steps to set the re-authentication number. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the port to be configured, and enter interface configuration mode.
Step 3	**dot1x max-reauth-req** *count*	Set the number of times that the switch restarts the authentication process before the port changes to the unauthorized state. The range is 1 to 10; the default is 2.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show dot1x interface** *interface-id*	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default re-authentication number, use the **no dot1x max-reauth-req** interface configuration command.

This example shows how to set 4 as the number of times that the switch restarts the authentication process before the port changes to the unauthorized state:

```
Switch(config-if)# dot1x max-reauth-req 4
```

Configuring the Host Mode

Beginning in privileged EXEC mode, follow these steps to allow multiple hosts (clients) on an IEEE 802.1x-authorized port that has the **dot1x port-control** interface configuration command set to **auto**. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the port to which multiple hosts are indirectly attached, and enter interface configuration mode.
Step 3	**dot1x host-mode multi-host**	Allow multiple hosts (clients) on an IEEE 802.1x-authorized port. Make sure that the **dot1x port-control** interface configuration command set is set to **auto** for the specified interface.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show dot1x interface** *interface-id*	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable multiple hosts on the port, use the **no dot1x host-mode multi-host** interface configuration command.

This example shows how to enable IEEE 802.1x and to allow multiple hosts:

```
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# dot1x port-control auto
Switch(config-if)# dot1x host-mode multi-host
```

Resetting the IEEE 802.1x Configuration to the Default Values

Beginning in privileged EXEC mode, follow these steps to reset the IEEE 802.1x configuration to the default values. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and specify the port to be configured.
Step 3	**dot1x default**	Reset the configurable IEEE 802.1x parameters to the default values.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show dot1x interface** *interface-id*	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Configuring IEEE 802.1x Accounting

Enabling AAA system accounting with IEEE 802.1x accounting allows system reload events to be sent to the accounting RADIUS server for logging. The server can then infer that all active IEEE 802.1x sessions are closed.

Because RADIUS uses the unreliable UDP transport protocol, accounting messages might be lost due to poor network conditions. If the switch does not receive the accounting response message from the RADIUS server after a configurable number of retransmissions of an accounting request, this system message appears:

```
Accounting message %s for session %s failed to receive Accounting Response.
```

When the stop message is not sent successfully, this message appears:

```
00:09:55: %RADIUS-3-NOACCOUNTINGRESPONSE: Accounting message Start for session
172.20.50.145 sam 11/06/03 07:01:16 11000002 failed to receive Accounting Response.
```

Note You must configure the RADIUS server to perform accounting tasks, such as logging start, stop, and interim-update messages and time stamps. To turn on these functions, enable logging of "Update/Watchdog packets from this AAA client" in your RADIUS server Network Configuration tab. Next, enable "CVS RADIUS Accounting" in your RADIUS server System Configuration tab.

Beginning in privileged EXEC mode, follow these steps to configure IEEE 802.1x accounting after AAA is enabled on your switch. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the port to be configured, and enter interface configuration mode.
Step 3	**aaa accounting dot1x default start-stop group radius**	Enable IEEE 802.1x accounting using the list of all RADIUS servers.
Step 4	**aaa accounting system default start-stop group radius**	(Optional) Enables system accounting (using the list of all RADIUS servers) and generates system accounting reload event messages when the switch reloads.
Step 5	**end**	Return to privileged EXEc mode.
Step 6	**show running-config**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Use the **show radius statistics** privileged EXEC command to display the number of RADIUS messages that do not receive the accounting response message.

This example shows how to configure IEEE 802.1x accounting. The first command configures the RADIUS server, specifying 1813 as the UDP port for accounting:

```
Switch(config)# radius-server host 172.120.39.46 auth-port 1812 acct-port 1813 key rad123
Switch(config)# aaa accounting dot1x default start-stop group radius
Switch(config)# aaa accounting system default start-stop group radius
```

Displaying IEEE 802.1x Statistics and Status

To display IEEE 802.1x statistics for all ports, use the **show dot1x all statistics** privileged EXEC command. To display IEEE 802.1x statistics for a specific port, use the **show dot1x statistics interface** *interface-id* privileged EXEC command.

To display the IEEE 802.1x administrative and operational status for the switch, use the **show dot1x all** privileged EXEC command. To display the IEEE 802.1x administrative and operational status for a specific port, use the **show dot1x interface** *interface-id* privileged EXEC command.

For detailed information about the fields in these displays, see the command reference for this release.



Configuring Interface Characteristics

This chapter defines the types of interfaces on the Cisco ME 3400 Ethernet Access switch and describes how to configure them.

The chapter consists of these sections:

- Understanding Interface Types, page 9-1
- Using Interface Configuration Mode, page 9-7
- Configuring Ethernet Interfaces, page 9-11
- Configuring Layer 3 Interfaces, page 9-20
- Configuring the System MTU, page 9-21
- Monitoring and Maintaining the Interfaces, page 9-22

Note For complete syntax and usage information for the commands used in this chapter, see the switch command reference for this release and the online *Cisco IOS Interface Command Reference, Release 12.2.*

Understanding Interface Types

This section describes the different types of interfaces supported by the switch with references to chapters that contain more detailed information about configuring these interface types. The rest of the chapter describes configuration procedures for physical interface characteristics.

These sections describe the interface types:

- Port-Based VLANs, page 9-2
- Switch Ports, page 9-2
- UNI and NNI Ports, page 9-3
- Routed Ports, page 9-4
- Switch Virtual Interfaces, page 9-5
- EtherChannel Port Groups, page 9-6
- Connecting Interfaces, page 9-6

Port-Based VLANs

A VLAN is a switched network that is logically segmented by function, team, or application, without regard to the physical location of the users. For more information about VLANs, see Chapter 11, "Configuring VLANs." Packets received on a port are forwarded only to ports that belong to the same VLAN as the receiving port. Network devices in different VLANs cannot communicate with one another without a Layer 3 device to route traffic between the VLANs.

VLAN partitions provide hard firewalls for traffic in the VLAN, and each VLAN has its own MAC address table. A VLAN comes into existence when a local port is configured to be associated with the VLAN or when a user creates a VLAN.

To isolate VLANs of different customers in a service-provider network, the Cisco ME switch uses UNI VLANs. UNI VLANs isolate user network interfaces (UNIs) on the switch from UNIs that belong to other customer VLANs:

- UNI isolated VLAN—This is the default VLAN state for all VLANs created on the switch. Local switching does not occur among UNIs on the switch that belong to the same UNI isolated VLAN.
- UNI community VLAN—Local switching is allowed among UNIs on the switch that belong to the same UNI community VLAN. If UNIs belong to the same customer, and you want to switch packets between the ports, you can configure the common VLAN as a UNI community VLAN.

For more information about UNI VLANs, see the "UNI VLANs" section on page 11-5.

To configure VLANs, use the **vlan** *vlan-id* global configuration command to enter VLAN configuration mode. The VLAN configurations for VLAN IDs 1 to 1005 are saved in the VLAN database. Extended-range VLANs (VLAN IDs 1006 to 4094) are not added to the VLAN database. VLAN configuration is saved in the switch running configuration, and you can save it in the switch startup configuration file by entering the **copy running-config startup-config** privileged EXEC command.

Add ports to a VLAN by using the **switchport** interface configuration commands:

- Identify the interface.
- For a trunk port, set trunk characteristics, and if desired, define the VLANs to which it can belong.
- For an access port, set and define the VLAN to which it belongs.
- For a tunnel port, set and define the VLAN ID for the customer-specific VLAN tag. See Chapter 13, "Configuring IEEE 802.1Q and Layer 2 Protocol Tunneling."

Switch Ports

Switch ports are Layer 2 only interfaces associated with a physical port. Switch ports belong to one or more VLANs. A switch port can be an access port, a trunk port, a private-VLAN port, or a tunnel port. You can configure a port as an access port or trunk port. You configure a private VLAN port as a host or promiscuous port that belongs to a private-VLAN primary or secondary VLAN. (Only NNIs can be configured as promiscuous ports.) You must manually configure tunnel ports as part of an asymmetric link connected to an IEEE 802.1Q trunk port. Switch ports are used for managing the physical interface and associated Layer 2 protocols and do not handle routing or bridging.

Configure switch ports by using the **switchport** interface configuration commands. Use the **switchport** command with no keywords to put an interface that is in Layer 3 mode into Layer 2 mode.

Note When you put an interface that is in Layer 3 mode into Layer 2 mode, the previous configuration information related to the affected interface might be lost, and the interface is returned to its default configuration.

For detailed information about configuring access port and trunk port characteristics, see Chapter 11, "Configuring VLANs." For more information about tunnel ports, see Chapter 13, "Configuring IEEE 802.1Q and Layer 2 Protocol Tunneling."

UNI and NNI Ports

User-network interfaces (UNIs) and network node interfaces (NNIs) are supported on the Cisco ME switch. UNIs are typically connected to a host, such as a PC or a Cisco IP phone. NNIs are typically connected to a router or to another switch. By default, the 10/100 ports on the Cisco ME switch are configured as UNIs, and the SFP module uplink ports are configured as NNIs.

The default status for a UNI is administratively down to prevent unauthorized users from gaining access to other ports as you configure the switch. Traffic is not switched between UNIs, and all arriving traffic at UNIs must leave on NNIs to prevent a user from gaining access to another user's private network. If it is appropriate for two or more UNIs to exchange traffic within the switch, the UNI can be assigned to a community VLAN. See Chapter 11, "Configuring VLANs," for instructions on how to configure community VLANs.

Note The exception to UNI default behavior is with IEEE 802.1Q tunneled traffic. A user can have multiple connections to a single switch by using IEEE 802.1Q tunneling, and the UNIs on the switch do not block the traffic by default.

Even though the default state for a UNI is shutdown, entering the **default interface** <*interface-id*> command changes the port to the enabled state.

The default status for an NNI is administratively up to allow a service provider remote access to the switch during initial configuration.

A port can be reconfigured from UNI to NNI and the reverse. When a port is reconfigured as another interface type, it inherits all the characteristics of that interface type. At any time, all ports on the Cisco ME switch are either UNI or NNI.

Access Ports

An access port belongs to and carries the traffic of only one VLAN. Traffic is received and sent in native formats with no VLAN tagging. Traffic arriving on an access port is assumed to belong to the VLAN assigned to the port. If an access port receives an IEEE 802.1Q tagged packet, the packet is dropped, and the source address is not learned. IEEE 802.1x can also be used for VLAN assignment.

Two types of access ports are supported:

- Static access ports are manually assigned to a VLAN.
- VLAN membership of dynamic access ports is learned through incoming packets. By default, a dynamic access port is a member of no VLAN, and forwarding to and from the port is enabled only when the VLAN membership of the port is discovered. UNIs begin forwarding packets as soon as

they are enabled. Dynamic access ports on the switch are assigned to a VLAN by a VLAN Membership Policy Server (VMPS). The VMPS can be a Catalyst 6500 series switch; the Cisco ME switch cannot be a VMPS server. Dynamic access ports for VMPS are only supported on UNIs.

Trunk Ports

An IEEE 802.1Q trunk port carries the traffic of multiple VLANs and by default is a member of all VLANs in the VLAN database. A trunk port supports simultaneous tagged and untagged traffic. An IEEE 802.1Q trunk port is assigned a default Port VLAN ID (PVID), and all untagged traffic travels on the port default PVID. All untagged traffic and tagged traffic with a NULL VLAN ID are assumed to belong to the port default PVID. A packet with a VLAN ID equal to the outgoing port default PVID is sent untagged. All other traffic is sent with a VLAN tag.

Although by default a trunk port is a member of multiple VLANs, you can limit VLAN membership by configuring an allowed list of VLANs for each trunk port. The list of allowed VLANs does not affect any other port but the associated trunk port. By default, all possible VLANs (VLAN ID 1 to 4094) are in the allowed list. A trunk port can become a member of a VLAN only if the VLAN is in the enabled state.

For more information about trunk ports, see Chapter 11, “Configuring VLANs.”

Tunnel Ports

Tunnel ports are used in IEEE 802.1Q tunneling to segregate the traffic of customers in a service-provider network from other customers who are using the same VLAN number. You configure an asymmetric link from a tunnel port on a service-provider edge switch to an IEEE 802.1Q trunk port on the customer switch. Packets entering the tunnel port on the edge switch, already IEEE 802.1Q-tagged with the customer VLANs, are encapsulated with another layer of an IEEE 802.1Q tag (called the metro tag), containing a VLAN ID unique in the service-provider network, for each customer. The double-tagged packets go through the service-provider network keeping the original customer VLANs separate from those of other customers. At the outbound interface, also a tunnel port, the metro tag is removed, and the original VLAN numbers from the customer network are retrieved.



Note IEEE 802.1Q tunneling is only supported when the switch is running the metro IP access or metro access image.

Tunnel ports cannot be trunk ports or access ports and must belong to a VLAN unique to each customer.

For more information about tunnel ports, see Chapter 13, “Configuring IEEE 802.1Q and Layer 2 Protocol Tunneling.”

Routed Ports

A routed port is a physical port that acts like a port on a router; it does not have to be connected to a router. A routed port is not associated with a particular VLAN, as is an access port. A routed port behaves like a regular router interface, except that it does not support VLAN subinterfaces. Routed ports can be configured with a Layer 3 routing protocol. A routed port is a Layer 3 interface only and does not support Layer 2 protocols, such as STP.

Configure routed ports by putting the interface into Layer 3 mode with the **no switchport** interface configuration command. Then assign an IP address to the port, enable routing, and assign routing protocol characteristics by using the **ip routing** and **router** *protocol* global configuration commands.

Note Entering a **no switchport** interface configuration command shuts down the interface and then re-enables it, which might generate messages on the device to which the interface is connected. When you put an interface that is in Layer 2 mode into Layer 3 mode, the previous configuration information related to the affected interface might be lost.

The number of routed ports that you can configure is not limited by software. However, the interrelationship between this number and the number of other features being configured might impact CPU performance because of hardware limitations. See the "Configuring Layer 3 Interfaces" section on page 9-20 for information about what happens when hardware resource limitations are reached.

For more information about IP unicast and multicast routing and routing protocols, see Chapter 32, "Configuring IP Unicast Routing" and Chapter 34, "Configuring IP Multicast Routing."

Note For full Layer 3 routing, you must have the metro IP access image installed on the switch

Switch Virtual Interfaces

A switch virtual interface (SVI) represents a VLAN of switch ports as one interface to the routing or bridging function in the system. Only one SVI can be associated with a VLAN, but you need to configure an SVI for a VLAN only when you wish to route between VLANs or to provide IP host connectivity to the switch. By default, an SVI is created for the default VLAN (VLAN 1) to permit remote switch administration. Additional SVIs must be explicitly configured.

Note You cannot delete interface VLAN 1.

SVIs provide IP host connectivity only to the system; in Layer 3 mode, you can configure routing across SVIs.

Although the switch supports a total of 1005 VLANs (and SVIs), the interrelationship between the number of SVIs and routed ports and the number of other features being configured might impact CPU performance because of hardware limitations. See the "Configuring Layer 3 Interfaces" section on page 9-20 for information about what happens when hardware resource limitations are reached.

SVIs are created the first time that you enter the **vlan** interface configuration command for a VLAN interface. The VLAN corresponds to the VLAN tag associated with data frames on an IEEE 802.1Q encapsulated trunk or the VLAN ID configured for an access port. Configure a VLAN interface for each VLAN for which you want to route traffic, and assign it an IP address. For more information, see the "Manually Assigning IP Information" section on page 3-9.

Note When you create an SVI, it does not become active until it is associated with a physical port.

SVIs support routing protocols. For more information about configuring IP routing, see Chapter 32, "Configuring IP Unicast Routing," and Chapter 34, "Configuring IP Multicast Routing."

Note Routed ports (or SVIs) are supported only when the metro IP access image is installed on the switch.

EtherChannel Port Groups

EtherChannel port groups treat multiple switch ports as one switch port. These port groups act as a single logical port for high-bandwidth connections between switches or between switches and servers. An EtherChannel balances the traffic load across the links in the channel. If a link within the EtherChannel fails, traffic previously carried over the failed link changes to the remaining links. You can group multiple trunk ports into one logical trunk port, group multiple access ports into one logical access port, group multiple tunnel ports into one logical tunnel port, or group multiple routed ports into one logical routed port. Most protocols operate over either single ports or aggregated switch ports and do not recognize the physical ports within the port group. Exceptions are the Cisco Discovery Protocol (CDP), Link Aggregation Control Protocol (LACP), and the Port Aggregation Protocol (PAgP), which operate only on physical NNI ports.

When you configure an EtherChannel, you create a port-channel logical interface and assign an interface to the EtherChannel. For Layer 3 interfaces, you manually create the logical interface by using the **interface port-channel** global configuration command. Then you manually assign an interface to the EtherChannel by using the **channel-group** interface configuration command. For Layer 2 interfaces, use the **channel-group** interface configuration command to dynamically create the port-channel logical interface. This command binds the physical and logical ports together. For more information, see Chapter 31, “Configuring EtherChannels.”

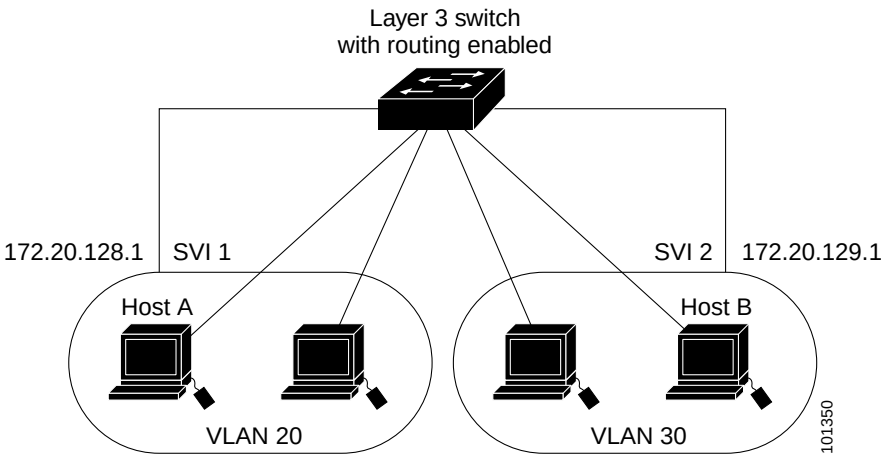
Connecting Interfaces

Devices within a single VLAN can communicate directly through any switch. Ports in different VLANs cannot exchange data without going through a routing device. With a standard Layer 2 switch, ports in different VLANs have to exchange information through a router.

By default, the Cisco ME switch provides VLAN isolation between UNIs. UNIs cannot exchange traffic unless they are changed to NNIs or assigned to a UNI community VLAN.

By using the switch with routing enabled, when you configure both VLAN 20 and VLAN 30 with an SVI to which an IP address is assigned, packets can be sent from Host A to Host B directly through the switch with no need for an external router (Figure 9-1).

Figure 9-1 Connecting VLANs with the Switch



When the metro IP access image is running on the switch, routing can be enabled on the switch. Whenever possible, to maintain high performance, forwarding is done by the switch hardware. However, only IP Version 4 packets with Ethernet II encapsulation can be routed in hardware. The routing function can be enabled on all SVIs and routed ports. The switch routes only IP traffic. When IP routing protocol parameters and address configuration are added to an SVI or routed port, any IP traffic received from these ports is routed. For more information, see Chapter 32, "Configuring IP Unicast Routing," Chapter 34, "Configuring IP Multicast Routing," and Chapter 35, "Configuring MSDP."

Using Interface Configuration Mode

The switch supports these interface types:

- Physical ports—switch ports, routed ports, UNIs, and NNIs
- VLANs—switch virtual interfaces
- Port-channels—EtherChannel interfaces

You can also configure a range of interfaces (see the "Configuring a Range of Interfaces" section on page 9-8).

To configure a physical interface (port), specify the interface type, the module number, and the switch port number, and enter interface configuration mode.

- Type—Fast Ethernet (fastethernet or fa) for 10/100 Mbps Ethernet, Gigabit Ethernet (gigabitethernet or gi) for 10/100/1000 Mbps Ethernet ports, or small form-factor pluggable (SFP) module Gigabit Ethernet interfaces.
- Module number—The module or slot number on the switch (always 0 on the Cisco ME switch).
- Port number—The interface number on the switch. The port numbers always begin at 1, starting with the leftmost port when facing the front of the switch, for example, fastethernet 0/1 or gigabitethernet 0/1. If there is more than one interface type (for example, 10/100 ports and SFP module ports), the port numbers restart with the second interface type: gigabitethernet 0/1.

You can identify physical interfaces by physically checking the interface location on the switch. You can also use the **show** privileged EXEC commands to display information about a specific interface or all the interfaces on the switch. The remainder of this chapter primarily provides physical interface configuration procedures.

Procedures for Configuring Interfaces

These general instructions apply to all interface configuration processes.

Step 1 Enter the **configure terminal** command at the privileged EXEC prompt:

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)#
```

Step 2 Enter the **interface** global configuration command. Identify the interface type and the number of the connector. In this example, Fast Ethernet port 1 is selected:

```
Switch(config)# interface fastethernet0/1
Switch(config-if)#
```

Note You do not need to add a space between the interface type and interface number. For example, in the preceding line, you can specify either **fastethernet 0/1**, **fastethernet0/1**, **fa 0/1**, or **fa0/1**.

Step 3 If you are configuring a UNI, enter the **no shutdown** interface configuration command to enable the interface:

```
Switch(config-if)# no shutdown
```

Step 4 Follow each **interface** command with the interface configuration commands that the interface requires. The commands that you enter define the protocols and applications that will run on the interface. The commands are collected and applied to the interface when you enter another interface command or enter **end** to return to privileged EXEC mode.

You can also configure a range of interfaces by using the **interface range** or **interface range macro** global configuration commands. Interfaces configured in a range must be the same type and must be configured with the same feature options.

Step 5 After you configure an interface, verify its status by using the **show** privileged EXEC commands listed in the "Monitoring and Maintaining the Interfaces" section on page 9-22.

Enter the **show interfaces** privileged EXEC command to see a list of all interfaces on or configured for the switch. A report is provided for each interface that the device supports or for the specified interface.

Configuring a Range of Interfaces

You can use the **interface range** global configuration command to configure multiple interfaces with the same configuration parameters. When you enter the interface range configuration mode, all command parameters that you enter are attributed to all interfaces within that range until you exit this mode.

Beginning in privileged EXEC mode, follow these steps to configure a range of interfaces with the same parameters:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface range** {*port-range*}	Specify the range of interfaces (VLANs or physical ports) to be configured, and enter interface range configuration mode. • You can use the **interface range** command to configure up to five port ranges or a previously defined macro. • In a comma-separated *port-range*, you must enter the interface type for each entry and enter spaces before and after the comma. • In a hyphen-separated *port-range*, you do not need to re-enter the interface type, but you must enter a space before the hyphen.
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled, and NNIs are enabled.
Step 4		You can now use the normal configuration commands to apply the configuration parameters to all interfaces in the range.
Step 5	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 6	**show interfaces** [*interface-id*]	Verify the configuration of the interfaces in the range.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

When using the **interface range** global configuration command, note these guidelines:

- Valid entries for *port-range*:
 - **vlan** *vlan-ID - vlan-ID*, where the VLAN ID is 1 to 4094
 - **fastethernet** module/{first *port*} - {*last port*}, where the module is always 0
 - **gigabitethernet** module/{*first port*} - {*last port*}, where the module is always 0
 - **port-channel** *port-channel-number - port-channel-number*, where the *port-channel-number* is 1 to 48

 Note When you use the **interface range** command with port channels, the first and last port channel number must be active port channels.

- The **interface range** command only works with VLAN interfaces that have been configured with the **interface vlan** command. The **show running-config** privileged EXEC command displays the configured VLAN interfaces. VLAN interfaces not displayed by the **show running-config** command cannot be used with the **interface range** command.
- All interfaces defined as in a range must be the same type (all Fast Ethernet ports, all Gigabit Ethernet ports, all EtherChannel ports, or all VLANs), but you can enter multiple ranges in a command.

This example shows how to use the **interface range** global configuration command to set the speed on ports 1 and 2 to 100 Mbps:

```
Switch# configure terminal
Switch(config)# interface range fastethernet0/1 - 2
Switch(config-if-range)# no shutdown
Switch(config-if-range)# speed 100
```

This example shows how to use a comma to add different interface type strings to the range to enable Fast Ethernet ports 1 to 3 and Gigabit Ethernet ports 1 and 2 to receive IEEE 802.3x flow control pause frames:

```
Switch# configure terminal
Switch(config)# interface range fastethernet0/1 - 3 , gigabitethernet0/1 - 2
Switch(config-if-range)# flowcontrol receive on
```

If you enter multiple configuration commands while you are in interface range mode, each command is executed as it is entered. The commands are not batched together and executed after you exit interface range mode. If you exit interface range configuration mode while the commands are being executed, some commands might not be executed on all interfaces in the range. Wait until the command prompt reappears before exiting interface range configuration mode.

Configuring and Using Interface Range Macros

You can create an interface range macro to automatically select a range of interfaces for configuration. Before you can use the **macro** keyword in the **interface range macro** global configuration command string, you must use the **define interface-range** global configuration command to define the macro.

Beginning in privileged EXEC mode, follow these steps to define an interface range macro:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**define interface-range** *macro_name interface-range*	Define the interface-range macro, and save it in NVRAM. • The *macro_name* is a 32-character maximum character string. • A macro can contain up to five comma-separated interface ranges. • Each *interface-range* must consist of the same port type.
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled, and NNIs are enabled.
Step 4	**interface range macro** *macro_name*	Select the interface range to be configured using the values saved in the interface-range macro called *macro_name.* You can now use the normal configuration commands to apply the configuration to all interfaces in the defined macro.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config \| include define**	Show the defined interface range macro configuration.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no define interface-range** *macro_name* global configuration command to delete a macro.

When using the **define interface-range** global configuration command, note these guidelines:

- Valid entries for *interface-range*:
 - **vlan** *vlan-ID - vlan-ID*, where the VLAN ID is 1 to 4094
 - **fastethernet** module/{first *port*} - {*last port*}, where the module is always 0
 - **gigabitethernet** module/{*first port*} - {*last port*}, where the module is always 0
 - **port-channel** *port-channel-number - port-channel-number*, where the *port-channel-number* is 1 to 48.

 Note When you use the interface ranges with port channels, the first and last port channel number must be active port channels.

- You must add a space between the first interface number and the hyphen when entering an *interface-range*. For example, **gigabitethernet0/1 - 2** is a valid range; **gigabitethernet0/1-2** is not a valid range.
- The VLAN interfaces must have been configured with the **interface vlan** command. The **show running-config** privileged EXEC command displays the configured VLAN interfaces. VLAN interfaces not displayed by the **show running-config** command cannot be used as *interface-ranges*.

- All interfaces defined as in a range must be the same type (all Fast Ethernet ports, all Gigabit Ethernet ports, all EtherChannel ports, or all VLANs), but you can combine multiple interface types in a macro.

This example shows how to define an interface-range named *enet_list* to include ports 1 and 2 and to verify the macro configuration:

```
Switch# configure terminal
Switch(config)# define interface-range enet_list gigabitethernet0/1 - 2
Switch(config)# end
Switch# show running-config | include define
define interface-range enet_list GigabitEthernet0/1 - 2
```

This example shows how to create a multiple-interface macro named *macro1* and assign all of the interfaces in the range to a VLAN:

```
Switch# configure terminal
Switch(config)# define interface-range macro1 fastethernet0/1 - 2, gigabitethernet0/1 - 2
Switch(config)# interface range macro macro1
Switch(config-if-range)# switchport access vlan 20
Switch(config-if-range)# no shut
Switch(config-if-range)# end
```

This example shows how to enter interface range configuration mode for the interface-range macro *enet_list*:

```
Switch# configure terminal
Switch(config)# interface range macro enet_list
Switch(config-if-range)#
```

This example shows how to delete the interface-range macro *enet_list* and to verify that it was deleted.

```
Switch# configure terminal
Switch(config)# no define interface-range enet_list
Switch(config)# end
Switch# show run | include define
Switch#
```

Configuring Ethernet Interfaces

These sections contain this configuration information:

- Default Ethernet Interface Configuration, page 9-12
- Configuring User Network and Network Node Interfaces, page 9-13
- Configuring Interface Speed and Duplex Mode, page 9-14
- Configuring IEEE 802.3x Flow Control, page 9-17
- Configuring Auto-MDIX on an Interface, page 9-18
- Adding a Description for an Interface, page 9-19

Default Ethernet Interface Configuration

Table 9-1 shows the Ethernet interface default configuration for NNIs, and Table 9-2 shows the Ethernet interface default configuration for UNIs. For more details on the VLAN parameters listed in the table, see Chapter 11, "Configuring VLANs." For details on controlling traffic to the port, see Chapter 21, "Configuring Port-Based Traffic Control."

Note To configure Layer 2 parameters, if the interface is in Layer 3 mode, you must enter the **switchport** interface configuration command without any parameters to put the interface into Layer 2 mode. This shuts down the interface and then re-enables it, which might generate messages on the device to which the interface is connected. When you put an interface that is in Layer 3 mode into Layer 2 mode, the previous configuration information related to the affected interface might be lost, and the interface is returned to its default configuration.

Table 9-1 Default Ethernet Configuration for NNIs

Feature	Default Setting
Operating mode	Layer 2 or switching mode (**switchport** command).
Allowed VLAN range	VLANs 1– 4094.
Default VLAN (for access ports)	VLAN 1 (Layer 2 interfaces only).
Native VLAN (for IEEE 802.1Q trunks)	VLAN 1 (Layer 2 interfaces only).
VLAN trunking	Switchport mode access (Layer 2 interfaces only).
Port enable state	Enabled.
Port description	None defined.
Speed	Autonegotiate.
Duplex mode	Autonegotiate.
IEEE 802.3x flow control	Flow control is set to **receive**: **off**. It is always off for sent packets.
EtherChannel	Disabled on all Ethernet ports. See Chapter 31, "Configuring EtherChannels."
Port blocking (unknown multicast and unknown unicast traffic)	Disabled (not blocked) (only Layer 2 interfaces). See the "Configuring Port Blocking" section on page 21-7.
Broadcast, multicast, and unicast storm control	Disabled. See the "Default Storm Control Configuration" section on page 21-3.
Port security	Disabled (only Layer 2 interfaces). See the "Default Port Security Configuration" section on page 21-10.
Port Fast	Disabled. See the "Default Optional Spanning-Tree Configuration" section on page 16-5.
Auto-MDIX	Enabled.
Cisco Discovery Protocol (CDP)	Enabled.
VMPS	Not configured.

Table 9-2 Default Ethernet Configuration for UNIs

Feature	Default Setting
Operating mode	Layer 2 or switching mode (**switchport** command).
Allowed VLAN range	VLANs 1– 4094.
Default VLAN (for access ports)	VLAN 1 (Layer 2 interfaces only).
Native VLAN (for IEEE 802.1Q trunks)	VLAN 1 (Layer 2 interfaces only).
VLAN trunking	Switchport mode access (Layer 2 interfaces only).
Dynamic VLAN	Enabled.
Port enable state	Disabled when no configuration file exists.
Port description	None defined.
Speed	Autonegotiate.
Duplex mode	Autonegotiate.
IEEE 802.3x flow control	Flow control is set to **receive**: **off**. It is always off for sent packets.
EtherChannel	Disabled on all Ethernet ports. See Chapter 31, "Configuring EtherChannels."
Port blocking (unknown multicast and unknown unicast traffic)	Disabled (not blocked) (only Layer 2 interfaces). See the "Configuring Port Blocking" section on page 21-7.
Broadcast, multicast, and unicast storm control	Disabled. See the "Default Storm Control Configuration" section on page 21-3.
Port security	Disabled (only Layer 2 interfaces). See the "Default Port Security Configuration" section on page 21-10.
Auto-MDIX	Enabled.

Configuring User Network and Network Node Interfaces

By default, all the 10/100 ports on the Cisco ME switch are configured as UNIs, and the SFP module ports are configured as NNIs.

You can use the **port-type** interface configuration command to change the port types. At any one time, only four ports on the switch can be configured as NNIs, but every port on the switch can be configured as UNIs.

When a port is changed from an NNI to a UNI, it inherits the configuration of the assigned VLAN, either in isolated or community mode. For more information about configuring UNI isolated and UNI community VLANs, see Chapter 11, "Configuring VLANs."

Note When you change a port from NNI to UNI or the reverse, any features exclusive to the port type are removed from the configuration to prevent conflicting configuration options on a specific interface type.

Beginning in privileged EXEC mode, follow these steps to configure the port type on an interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode
Step 2	**interface** *interface-id*	Specify the interface to configure, and enter interface configuration mode.
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled, and NNIs are enabled.
Step 4	**port-type** {**uni** \| **nni**}	Change a port to a UNI or an NNI.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show interfaces** *interface-id*	Verify the interface IEEE 802.3x flow control settings.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

This example shows how to change a port from a UNI to an NNI and save it to the running configuration.

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# interface fastethernet0/1
Switch(config-if)# port-type nni
Switch(config-if)# no shutdown
5d20h: %SYS-5-CONFIG_I: Configured from console by console
Switch(config-if)# end
Switch# copy running-config startup-config
```

Configuring Interface Speed and Duplex Mode

Ethernet interfaces on the switch operate at 10, 100, or 1000 Mbps and in either full- or half-duplex mode. In full-duplex mode, two stations can send and receive traffic at the same time. Normally, 10-Mbps ports operate in half-duplex mode, which means that stations can either receive or send traffic.

Switch models include combinations of Fast Ethernet (10/100-Mbps) ports, Gigabit Ethernet (10/100/1000-Mbps) ports, and small form-factor pluggable (SFP) module slots supporting SFP modules.

These sections describe how to configure the interface speed and duplex mode:

- Speed and Duplex Configuration Guidelines, page 9-15
- Setting the Interface Speed and Duplex Parameters, page 9-15

Speed and Duplex Configuration Guidelines

When configuring an interface speed and duplex mode, note these guidelines:

- You can configure interface speed on Fast Ethernet (10/100-Mbps) and Gigabit Ethernet (10/100/1000-Mbps) ports. You can configure Fast Ethernet ports to full-duplex, half-duplex, or to autonegotiate mode. You can configure Gigabit Ethernet ports to full-duplex mode or to autonegotiate. You also can configure Gigabit Ethernet ports to half-duplex mode if the speed is 10 or 100 Mbps. Half-duplex mode is not supported on Gigabit Ethernet ports operating at 1000 Mbps.
- With the exception of when 1000BASE-T SFP modules are installed in the SFP module slots, you cannot configure speed on SFP module ports, but you can configure speed to not negotiate (**nonegotiate**) if connected to a device that does not support autonegotiation.

 However, when a 1000BASE-T SFP module is in the SFP module slot, you can configure speed as 10, 100, or 1000 Mbps, or auto, but not as **nonegotiate**.

 On a 100BASE-FX SFP module, you cannot configure the speed as **nonegotiate**.
- You cannot configure duplex mode on SFP module ports; they operate in full-duplex mode except in these situations:
 - When a Cisco1000BASE-T SFP module is in the SFP module slot, you can configure duplex mode to **auto** or **full**. Half-duplex mode is supported with the **auto** setting.
 - When a Cisco100BASE-FX SFP module is in the SFP module slot, you can configure duplex mode to **half** or **full**. Although the **auto** keyword is available, it puts the interface in half-duplex mode (the default for this SFP module) because the 100BASE-FX SFP module does not support autonegotiation.
- If both ends of the line support autonegotiation, we highly recommend the default setting of **auto** negotiation.
- If one interface supports autonegotiation and the other end does not, configure duplex and speed on both interfaces; do not use the **auto** setting on the supported side.
- When STP is enabled and a port is reconfigured, the switch can take up to 30 seconds to check for loops. The port LED is amber while STP reconfigures. On the Cisco ME switch, STP is supported only on NNIs.

Caution Changing the interface speed and duplex mode configuration might shut down and re-enable the interface during the reconfiguration.

Setting the Interface Speed and Duplex Parameters

Beginning in privileged EXEC mode, follow these steps to set the speed and duplex mode for a physical interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the physical interface to be configured, and enter interface configuration mode.
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled, and NNIs are enabled.

	Command	Purpose
Step 4	**speed** {**10** \| **100** \| **1000** \| **auto** [**10** \| **100** \| **1000**] \| **nonegotiate**}	Enter the appropriate speed parameter for the interface: • Enter **10**, **100**, or **1000** to set a specific speed for the interface. The **1000** keyword is available only for 10/100/1000 Mbps ports or SFP module ports with a 1000BASE-T SFP module. • Enter **auto** to enable the interface to autonegotiate speed with the connected device. If you use the **10**, **100**, or the **1000** keywords with the **auto** keyword, the port autonegotiates only at the specified speeds. • The **nonegotiate** keyword is available only for SFP module ports. SFP module ports operate only at 1000 Mbps but can be configured to not negotiate if connected to a device that does not support autonegotiation. **Note** When a Cisco1000BASE-T SFP module is in the SFP module slot, the speed can be configured to **10**, **100**, **1000**, or to **auto**, but not to **nonegotiate**.
Step 5	**duplex** {**auto** \| **full** \| **half**}	Enter the duplex parameter for the interface. Enable half-duplex mode (for interfaces operating only at 10 or 100 Mbps). You cannot configure half-duplex mode for interfaces operating at 1000 Mbps. You can configure the duplex setting when the speed is set to **auto**. This command is not available on SFP module ports with these exceptions: • If a Cisco 1000BASE-T SFP module is inserted, you can configure duplex to **auto** or to **full**. • If a Cisco 100BASE-FX SFP module is inserted, you can configure duplex to **full** or to **half**. Although the **auto** keyword is available, it puts the interface in half-duplex mode (the default).
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show interfaces** *interface-id*	Display the interface speed and duplex mode configuration.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no speed** and **no duplex** interface configuration commands to return the interface to the default speed and duplex settings (autonegotiate). To return all interface settings to the defaults, use the **default interface** *interface-id* interface configuration command.

This example shows how to set the interface speed to 10 Mbps and the duplex mode to half on a 10/100 Mbps port:

```
Switch# configure terminal
Switch(config)# interface fasttethernet0/3
Switch(config-if)# no shutdown
Switch(config-if)# speed 10
Switch(config-if)# duplex half
```

This example shows how to set the interface speed to 100 Mbps on a 10/100/1000 Mbps port:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet0/2
Switch(config-if)# speed 100
```

Configuring IEEE 802.3x Flow Control

IEEE 802.3x flow control enables connected Ethernet ports to control traffic rates during congestion by allowing congested nodes to pause link operation at the other end. If one port experiences congestion and cannot receive any more traffic, it notifies the other port by sending a pause frame to stop sending until the condition clears. Upon receipt of a pause frame, the sending device stops sending any data packets, which prevents any loss of data packets during the congestion period.

Note Ports can receive, but not send, pause frames.

You use the **flowcontrol** interface configuration command to set the interface's ability to **receive** pause frames to **on**, **off**, or **desired**. The default state is **off**.

When set to **desired**, an interface can operate with an attached device that is required to send flow-control packets or with an attached device that is not required to but can send flow-control packets.

These rules apply to IEEE 802.3x flow control settings on the device:

- **receive on** (or **desired**): The port cannot send pause frames but can operate with an attached device that is required to or can send pause frames; the port can receive pause frames.
- **receive off**: IEEE 802.3x flow control does not operate in either direction. In case of congestion, no indication is given to the link partner, and no pause frames are sent or received by either device.

Note For details on the command settings and the resulting IEEE 802.3x flow control resolution on local and remote ports, see the **flowcontrol** interface configuration command in the command reference for this release.

Beginning in privileged EXEC mode, follow these steps to configure IEEE 802.3x flow control on an interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode
Step 2	**interface** *interface-id*	Specify the physical interface to be configured, and enter interface configuration mode.
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled, and NNIs are enabled.
Step 4	**flowcontrol** {**receive**} {**on** \| **off** \| **desired**}	Configure the IEEE 802.3x flow control mode for the port.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show interfaces** *interface-id*	Verify the interface IEEE 802.3x flow control settings.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable IEEE 802.3x flow control, use the **flowcontrol receive off** interface configuration command.

This example shows how to enable IEEE 802.3x flow control on a port:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# flowcontrol receive on
Switch(config-if)# end
```

Configuring Auto-MDIX on an Interface

When automatic medium-dependent interface crossover (auto-MDIX) is enabled on an interface, the interface automatically detects the required cable connection type (straight through or crossover) and configures the connection appropriately. When connecting switches without the auto-MDIX feature, you must use straight-through cables to connect to devices such as servers, workstations, or routers and crossover cables to connect to other switches or repeaters. With auto-MDIX enabled, you can use either type of cable to connect to other devices, and the interface automatically corrects for any incorrect cabling. For more information about cabling requirements, see the hardware installation guide.

Auto-MDIX is enabled by default. When you enable auto-MDIX, you must also set the speed and duplex on the interface to **auto** so that the feature operates correctly. Auto-MDIX is supported on all 10/100 and 10/100/1000 Mbps interfaces and on Cisco 10/100/1000 BASE-T/TX SFP module interfaces. It is not supported on 1000 BASE-SX or -LX SFP module interfaces.

Table 9-3 shows the link states that result from auto-MDIX settings and correct and incorrect cabling.

Table 9-3 Link Conditions and Auto-MDIX Settings

Local Side Auto-MDIX	Remote Side Auto-MDIX	With Correct Cabling	With Incorrect Cabling
On	On	Link up	Link up
On	Off	Link up	Link up
Off	On	Link up	Link up
Off	Off	Link up	Link down

Beginning in privileged EXEC mode, follow these steps to configure auto-MDIX on an interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode
Step 2	**interface** *interface-id*	Specify the physical interface to be configured, and enter interface configuration mode.
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled, and NNIs are enabled.
Step 4	**speed auto**	Configure the interface to autonegotiate speed with the connected device.
Step 5	**duplex auto**	Configure the interface to autonegotiate duplex mode with the connected device.
Step 6	**mdix auto**	Enable auto-MDIX on the interface.
Step 7	**end**	Return to privileged EXEC mode.
Step 8	**show controllers ethernet-controller** *interface-id* **phy**	Verify the operational state of the auto-MDIX feature on the interface.
Step 9	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable auto-MDIX, use the **no mdix auto** interface configuration command.

This example shows how to enable auto-MDIX on a port:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# no shutdown
Switch(config-if)# speed auto
Switch(config-if)# duplex auto
Switch(config-if)# mdix auto
Switch(config-if)# end
```

Adding a Description for an Interface

You can add a description about an interface to help you remember its function. The description appears in the output of these privileged EXEC commands: **show configuration**, **show running-config**, and **show interfaces**.

Beginning in privileged EXEC mode, follow these steps to add a description for an interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface for which you are adding a description, and enter interface configuration mode.
Step 3	**description** *string*	Add a description (up to 240 characters) for an interface.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show interfaces** *interface-id* **description** or **show running-config**	Verify your entry.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no description** interface configuration command to delete the description.

This example shows how to add a description on a port and how to verify the description:

```
Switch# config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# interface gigabitethernet0/2
Switch(config-if)# description Connects to Marketing
Switch(config-if)# end
Switch# show interfaces gigabitethernet0/2 description
Interface Status         Protocol Description
Gi 0/2   admin down     down     Connects to Marketing
```

Configuring Layer 3 Interfaces

The Cisco 3400 ME switch must be running the metro IP access image to support Layer 3 interfaces. The Cisco ME switch supports these types of Layer 3 interfaces:

- SVIs: You should configure SVIs for any VLANs for which you want to route traffic. SVIs are created when you enter a VLAN ID following the **interface vlan** global configuration command. To delete an SVI, use the **no interface vlan** global configuration command. You cannot delete interface VLAN 1.

 Note When you create an SVI, it does not become active until it is associated with a physical port. For information about assigning Layer 2 ports to VLANs, see Chapter 11, "Configuring VLANs."

- Routed ports: Routed ports are physical ports configured to be in Layer 3 mode by using the **no switchport** interface configuration command.
- Layer 3 EtherChannel ports: EtherChannel interfaces made up of routed ports.

 EtherChannel port interfaces are described in Chapter 31, "Configuring EtherChannels."

A Layer 3 switch can have an IP address assigned to each routed port and SVI.

There is no defined limit to the number of SVIs and routed ports that can be configured in a switch. However, the interrelationship between the number of SVIs and routed ports and the number of other features being configured might have an impact on CPU usage because of hardware limitations. If the switch is using maximum hardware resources, attempts to create a routed port or SVI have these results:

- If you try to create a new routed port, the switch generates a message that there are not enough resources to convert the interface to a routed port, and the interface remains as a switch port.
- If you try to create an extended-range VLAN, an error message is generated, and the extended-range VLAN is rejected.
- If the switch attempts to boot up with a configuration that has more VLANs and routed ports than hardware can support, the VLANs are created, but the routed ports are shut down, and the switch sends a message that this was due to insufficient hardware resources.

All Layer 3 interfaces require an IP address to route traffic. This procedure shows how to configure an interface as a Layer 3 interface and how to assign an IP address to an interface.

Note If the physical port is in Layer 2 mode (the default), you must enter the **no switchport** interface configuration command to put the interface into Layer 3 mode. Entering a **no switchport** command disables and then re-enables the interface, which might generate messages on the device to which the interface is connected. Furthermore, when you put an interface that is in Layer 2 mode into Layer 3 mode, the previous configuration information related to the affected interface might be lost, and the interface is returned to its default configuration

Beginning in privileged EXEC mode, follow these steps to configure a Layer 3 interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** {{**fastethernet** \| **gigabitethernet**} *interface-id*} \| {**vlan** *vlan-id*} \| {**port-channel** *port-channel-number*}	Specify the interface to be configured as a Layer 3 interface, and enter interface configuration mode.

	Command	Purpose
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled, and NNIs are enabled.
Step 4	**no switchport**	For physical ports only, enter Layer 3 mode.
Step 5	**ip address** *ip_address subnet_mask*	Configure the IP address and IP subnet.
Step 6	**no shutdown**	Enable the interface.
Step 7	**end**	Return to privileged EXEC mode.
Step 8	**show interfaces** [*interface-id*] **show ip interface** [*interface-id*] **show running-config interface** [*interface-id*]	Verify the configuration.
Step 9	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove an IP address from an interface, use the **no ip address** interface configuration command.

This example shows how to configure a port as a routed port and to assign it an IP address:

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# interface gigabitethernet0/2
Switch(config-if)# no switchport
Switch(config-if)# ip address 192.20.135.21 255.255.255.0
```

Configuring the System MTU

The default maximum transmission unit (MTU) size for frames received and sent on all interfaces on the switch is 1500 bytes. You can increase the MTU size for all interfaces operating at 10 or 100 Mbps by using the **system mtu** global configuration command. You can increase the MTU size to support jumbo frames on all Gigabit Ethernet interfaces by using the **system mtu jumbo** global configuration command. Gigabit Ethernet ports are not affected by the **system mtu** command. Fast Ethernet ports are not affected by the **system jumbo mtu** command because jumbo frames are not supported on 10/100 interfaces, including 100BASE-FX and 100BASE-BX SFP modules.

You cannot set the MTU size for an individual interface; you set it for all 10/100 or all Gigabit Ethernet interfaces on the switch. When you change the MTU size, you must reset the switch before the new configuration takes effect.

The size of frames that can be received by the switch CPU is limited to 1546 bytes, no matter what value was entered with the **system mtu** or **system mtu jumbo** commands. On 10/100 interfaces, the maximum size of frames is limited to 1546 bytes. Although frames that are forwarded or routed typically are not received by the CPU, in some cases packets are sent to the CPU, such as traffic sent to control traffic, SNMP, Telnet, or routing protocols.

Routed packets are subjected to MTU checks on the sending ports. The MTU value used for routed ports is derived from the configured **system mtu** value (not the **system mtu jumbo** value). That is, the routed MTU is never greater than the system MTU for any VLAN. The routing protocols use the system MTU value when negotiating adjacencies and the MTU of the link. For example, the Open Shortest Path First (OSPF) protocol uses this MTU value before setting up an adjacency with a peer router. To view the MTU value for routed packets for a specific VLAN, use the **show platform port-asic mvid** privileged EXEC command.

Note If Layer 2 Gigabit Ethernet interfaces are configured to accept frames greater than the 10/100 interfaces, jumbo frames received on a Layer 2 Gigabit Ethernet interface and sent on a Layer 2 10/100 interface are dropped.

Beginning in privileged EXEC mode, follow these steps to change the MTU size for all 10/100 or Gigabit Ethernet interfaces:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**system mtu** *bytes*	(Optional) Change the MTU size for all interfaces on the switch that are operating at 10 or 100 Mbps. The range is 1500 to 1546 bytes; the default is 1500 bytes.
Step 3	**system mtu jumbo** *bytes*	(Optional) Change the MTU size for all Gigabit Ethernet interfaces on the switch. The range is 1500 to 9000 bytes; the default is 1500 bytes.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**copy running-config startup-config**	Save your entries in the configuration file.
Step 6	**reload**	Reload the operating system.

If you enter a value that is outside the allowed range for the specific type of interface, the value is not accepted.

After the switch reloads, you can verify your settings by entering the **show system mtu** privileged EXEC command.

This example shows how to set the maximum packet size for a Gigabit Ethernet port to 1800 bytes:

```
Switch(config)# system jumbo mtu 1800
Switch(config)# exit
Switch# reload
```

This example shows the response when you try to set Gigabit Ethernet interfaces to an out-of-range number:

```
Switch(config)# system mtu jumbo 25000
                                 ^
% Invalid input detected at '^' marker.
```

Monitoring and Maintaining the Interfaces

These sections contain interface monitoring and maintenance information:

- Monitoring Interface Status, page 9-23
- Clearing and Resetting Interfaces and Counters, page 9-24
- Shutting Down and Restarting the Interface, page 9-24

Monitoring Interface Status

Commands entered at the privileged EXEC prompt display information about the interface, including the versions of the software and the hardware, the configuration, and statistics about the interfaces. Table 9-4 lists some of these interface monitoring commands. (You can display the full list of **show** commands by using the **show ?** command at the privileged EXEC prompt.) These commands are fully described in the *Cisco IOS Interface Command Reference, Release 12.2.*

Table 9-4 Show Commands for Interfaces

Command	Purpose
show interfaces [*interface-id*]	Display the status and configuration of all interfaces or a specific interface.
show interfaces *interface-id* **status** [**err-disabled**]	Display interface status or a list of interfaces in an error-disabled state.
show interfaces [*interface-id*] **switchport**	Display administrative and operational status of switching mode. You can use this command to find out if a port is in routing or in switching mode.
show interfaces [*interface-id*] **description**	Display the description configured on an interface or all interfaces and the interface status.
show ip interface [*interface-id*]	Display the usability status of all interfaces configured for IP routing or the specified interface.
show interface [*interface-id*] **stats**	Display the input and output packets by the switching path for the interface.
show interfaces transceiver properties	(Optional) Display speed and duplex settings on the interface.
show interfaces transceiver properties	(Optional) Display temperature, voltage, or amount of current on the interface.
show interfaces [*interface-id*] [{**transceiver properties** \| **detail**}] *module number*]	Display physical and operational status about an SFP module.
show port-type [*uni* \| *nni*]	Display interface type information for the Cisco ME switch.
show running-config interface [*interface-id*]	Display the running configuration in RAM for the interface.
show version	Display the hardware configuration, software version, the names and sources of configuration files, and the boot images.
show controllers ethernet-controller *interface-id* **phy**	Display the operational state of the auto-MDIX feature on the interface.

Clearing and Resetting Interfaces and Counters

Table 9-5 lists the privileged EXEC mode **clear** commands that you can use to clear counters and reset interfaces.

Table 9-5 Clear Commands for Interfaces

Command	Purpose
clear counters [*interface-id*]	Clear interface counters.
clear interface *interface-id*	Reset the hardware logic on an interface.
clear line [*number* \| **console 0** \| **vty** *number*]	Reset the hardware logic on an asynchronous serial line.

To clear the interface counters shown by the **show interfaces** privileged EXEC command, use the **clear counters** privileged EXEC command. The **clear counters** command clears all current interface counters from the interface unless you specify optional arguments that clear only a specific interface type from a specific interface number.

Note The **clear counters** privileged EXEC command does not clear counters retrieved by using Simple Network Management Protocol (SNMP), but only those seen with the **show interface** privileged EXEC command.

Shutting Down and Restarting the Interface

Shutting down an interface disables all functions on the specified interface and marks the interface as unavailable on all monitoring command displays. This information is communicated to other network servers through all dynamic routing protocols. The interface is not mentioned in any routing updates.

Beginning in privileged EXEC mode, follow these steps to shut down an interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** {**vlan** *vlan-id*} \| {{**fastethernet** \| **gigabitethernet**} *interface-id*} \| {**port-channel** *port-channel-number*}	Select the interface to be configured.
Step 3	**shutdown**	Shut down an interface.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entry.

Use the **no shutdown** interface configuration command to restart the interface.

To verify that an interface is disabled, enter the **show interfaces** privileged EXEC command. A disabled interface is shown as *administratively down* in the display.



Configuring Command Macros

This chapter describes how to configure and apply command macros on the Cisco 3400 ME switch.

Note For complete syntax and usage information for the commands used in this chapter, see the command reference for this release.

This chapter consists of these sections:

- Understanding Command Macros, page 10-1
- Configuring Command Macros, page 10-2
- Displaying Command Macros, page 10-5

Understanding Command Macros

Command macros provide a convenient way to save and share common configurations. You can use command macros to enable features and settings based on the location of a switch in the network and for mass configuration deployments across the network.

Each command macro is a set of command-line interface (CLI) commands that you define. Command macros do not contain new CLI commands; they are simply a group of existing CLI commands.

When you apply a command macro on an interface, the CLI commands within the macro are configured on the interface. When the macro is applied to an interface, the existing interface configurations are not lost. The new commands are added to the interface and are saved in the running configuration file.

Cisco provides a collection of pretested, Cisco-recommended baseline configuration templates for Cisco ME switches. The online reference guide templates provide the CLI commands that you can use to create command macros based on the usage of the port. You can use the configuration templates to create command macros to build and deploy Cisco-recommended network designs and configurations. For more information about Cisco-recommended configuration templates, see this website:

http://www.cisco.com/go/smartports

Configuring Command Macros

You can create a new command macro or use an existing macro as a template to create a new macro that is specific to your application. After you create the macro, you can apply it globally to a switch, to a switch interface, or to a range of interfaces.

These sections contain this configuration information:

- Default Command Macro Configuration, page 10-2
- Command Macro Configuration Guidelines, page 10-2
- Creating Command Macros, page 10-3
- Applying Command Macros, page 10-4
- Displaying Command Macros, page 10-5

Default Command Macro Configuration

There are no command macros enabled.

Command Macro Configuration Guidelines

Follow these guidelines when configuring macros on your switch:

- When creating a macro, do not use the **exit** or **end** commands or change the command mode by using **interface** *interface-id*. This could cause commands that follow **exit**, **end**, or **interface** *interface-id* to execute in a different command mode.
- When creating a macro, all CLI commands should be in the same configuration mode.
- When creating a macro that requires the assignment of unique values, use the **parameter** *value* keywords to designate values specific to the interface. Keyword matching is case sensitive. All matching occurrences of the keyword are replaced with the corresponding value. Any full match of a keyword, even if it is part of a larger string, is considered a match and is replaced by the corresponding value.
- Macro names are case sensitive. For example, the commands **macro name Sample-Macro** and **macro name sample-macro** will result in two separate macros.
- Some macros might contain keywords that require a parameter value. You can use the **macro global apply** *macro-name* **?** global configuration command or the **macro apply** *macro-name* **?** interface configuration command to display a list of any required values in the macro. If you apply a macro without entering the keyword values, the commands are invalid and are not applied.
- When a macro is applied globally to a switch or to a switch interface, all existing configuration on the interface is retained. This is helpful when applying an incremental configuration.
- If you modify a macro definition by adding or deleting commands, the changes are not reflected on the interface where the original macro was applied. You need to reapply the updated macro on the interface to apply the new or changed commands.
- You can use the **macro global trace** *macro-name* global configuration command or the **macro trace** *macro-name* interface configuration command to apply and debug a macro to find any syntax or configuration errors. If a command fails because of a syntax error or a configuration error, the macro continues to apply the remaining commands.

- Some CLI commands are specific to certain interface types. If a macro is applied to an interface that does not accept the configuration, the macro will fail the syntax check or the configuration check, and the switch will return an error message.
- Applying a macro to an interface range is the same as applying a macro to a single interface. When you use an interface range, the macro is applied sequentially to each interface within the range. If a macro command fails on one interface, it is still applied to the remaining interfaces.
- When you apply a macro to a switch or a switch interface, the macro name is automatically added to the switch or interface. You can display the applied commands and macro names by using the **show running-config** user EXEC command.
- When you apply a macro to a UNI, you must first enable the port. UNIs are disabled by default.

Creating Command Macros

Beginning in privileged EXEC mode, follow these steps to create a command macro:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**macro name** *macro-name*	Create a macro definition, and enter a macro name. A macro definition can contain up to 3000 characters. Enter the macro commands with one command per line. Use the **@** character to end the macro. Use the **#** character at the beginning of a line to enter comment text within the macro. (Optional) You can define keywords within a macro by using a help string to specify the keywords. Enter **# macro keywords** *word* to define the keywords that are available for use with the macro. Separated by a space, you can enter up to three help string keywords in a macro. Macro names are case sensitive. For example, the commands **macro name Sample-Macro** and **macro name sample-macro** will result in two separate macros. We recommend that you do not use the **exit** or **end** commands or change the command mode by using **interface** *interface-id* in a macro. This could cause any commands following **exit**, **end**, or **interface** *interface-id* to execute in a different command mode. For best results, all commands in a macro should be in the same configuration mode.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show parser macro name** *macro-name*	Verify that the macro was created.

The **no** form of the **macro name** global configuration command only deletes the macro definition. It does not affect the configuration of those interfaces on which the macro is already applied.

This example shows how to create a macro that defines the switchport access VLAN and the number of secure MAC addresses and also includes two help string keywords by using **# macro keywords**:

```
Switch(config)# macro name test
switchport access vlan $VLANID
switchport port-security maximum $MAX
#macro keywords $VLANID $MAX
@
```

Applying Command Macros

Beginning in privileged EXEC mode, follow these steps to apply a command macro:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**macro global** {**apply** \| **trace**} *macro-name* [**parameter** {*value*}] [**parameter** {*value*}] [**parameter** {*value*}]	Apply each individual command defined in the macro to the switch by entering **macro global apply** *macro-name*. Specify **macro global trace** *macro-name* to apply and debug a macro to find any syntax or configuration errors. (Optional) Specify unique parameter values that are specific to the switch. You can enter up to three keyword-value pairs. Parameter keyword matching is case sensitive. All matching occurrences of the keyword are replaced with the corresponding value. Some macros might contain keywords that require a parameter value. You can use the **macro global apply** *macro-name* **?** command to display a list of any required values in the macro. If you apply a macro without entering the keyword values, the commands are invalid and are not applied.
Step 3	**macro global description** *text*	(Optional) Enter a description about the macro that is applied to the switch.
Step 4	**interface** *interface-id*	(Optional) Enter interface configuration mode, and specify the interface on which to apply the macro.
Step 5	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled, and NNIs are enabled.
Step 6	**default interface** *interface-id*	(Optional) Clear all configuration from the specified interface.
Step 7	**macro** {**apply** \| **trace**} *macro-name* [**parameter** {*value*}] [**parameter** {*value*}] [**parameter** {*value*}]	Apply each individual command defined in the macro to the interface by entering **macro apply** *macro-name*. Specify **macro trace** *macro-name* to apply and debug a macro to find any syntax or configuration errors. (Optional) Specify unique parameter values that are specific to the interface. You can enter up to three keyword-value pairs. Parameter keyword matching is case sensitive. All matching occurrences of the keyword are replaced with the corresponding value. Some macros might contain keywords that require a parameter value. You can use the **macro apply** *macro-name* **?** command to display a list of any required values in the macro. If you apply a macro without entering the keyword values, the commands are invalid and are not applied.
Step 8	**macro description** *text*	(Optional) Enter a description about the macro that is applied to the interface.
Step 9	**end**	Return to privileged EXEC mode.
Step 10	**show parser macro description** [**interface** *interface-id*]	Verify that the macro is applied to the interface.
Step 11	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

You can delete a global macro-applied configuration on a switch only by entering the **no** version of each command that is in the macro. You can delete a macro-applied configuration on an interface by entering the **default interface** *interface-id* interface configuration command.

This example shows how to apply the user-created macro called **snmp**, to set the hostname address to **test-server**, and to set the IP precedence value to **7**:

```
Switch(config)# macro global apply snmp ADDRESS test-server VALUE 7
```

This example shows how to debug the user-created macro called **snmp** by using the **macro global trace** global configuration command to find any syntax or configuration errors in the macro when it is applied to the switch.

```
Switch(config)# macro global trace snmp VALUE 7
Applying command...‘snmp-server enable traps port-security’
Applying command...‘snmp-server enable traps linkup’
Applying command...‘snmp-server enable traps linkdown’
Applying command...‘snmp-server host’
%Error Unknown error.
Applying command...‘snmp-server ip precedence 7’
```

This example shows how to apply the user-created macro called **desktop-config** and to verify the configuration.

```
Switch(config)# interface gigabitethernet0/2
Switch(config-if)# macro apply desktop-config
Switch(config-if)# end
Switch# show parser macro description
Interface    Macro Description
---------------------------------------------------------------
Gi0/2      desktop-config
---------------------------------------------------------------
```

This example shows how to apply the user-created macro called **desktop-config** and to replace all occurrences of VLAN 1 with VLAN 25:

```
Switch(config-if)# macro apply desktop-config vlan 25
```

Displaying Command Macros

To display the command macros, use one or more of the privileged EXEC commands in Table 10-1.

Table 10-1 Commands for Displaying Command Macros

Command	Purpose
show parser macro	Displays all configured macros.
show parser macro name *macro-name*	Displays a specific macro.
show parser macro brief	Displays the configured macro names.
show parser macro description [**interface** *interface-id*]	Displays the macro description for all interfaces or for a specified interface.



CHAPTER 11

Configuring VLANs

This chapter describes how to configure normal-range VLANs (VLAN IDs 1 to 1005) and extended-range VLANs (VLAN IDs 1006 to 4094) on the Cisco ME 3400 Ethernet Access switch. It includes information about VLAN membership modes, VLAN configuration modes, VLAN trunks, and dynamic VLAN assignment from a VLAN Membership Policy Server (VMPS).

Note For complete syntax and usage information for the commands used in this chapter, see the command reference for this release.

The chapter consists of these sections:

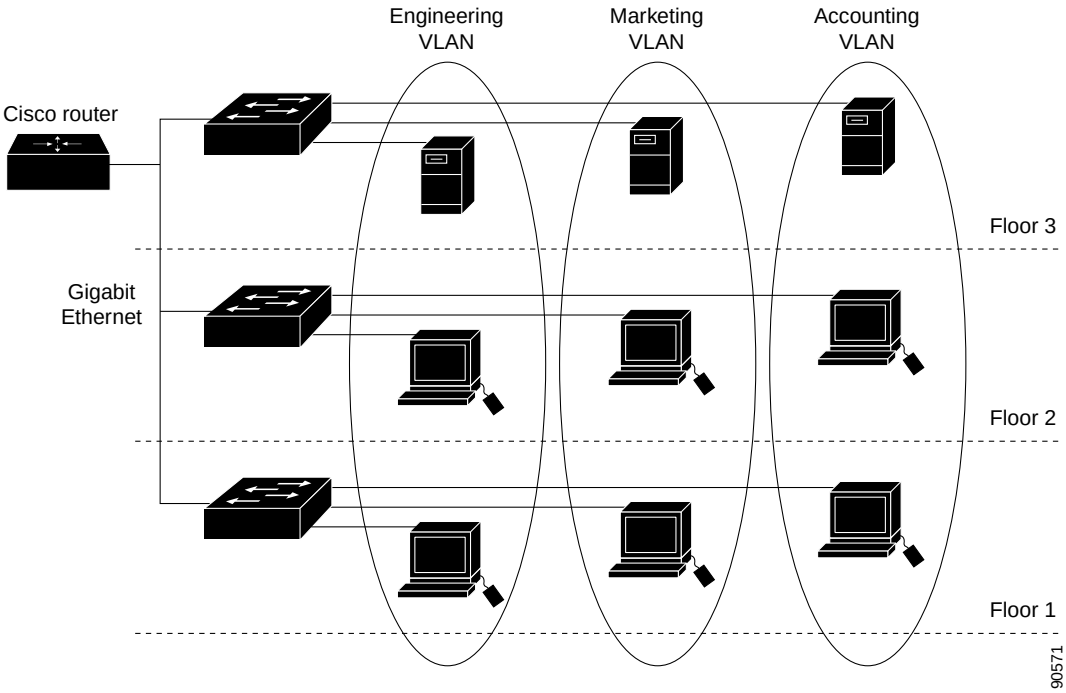
- Understanding VLANs, page 11-1
- Creating and Modifying VLANs, page 11-6
- Displaying VLANs, page 11-14
- Configuring VLAN Trunks, page 11-14
- Configuring VMPS, page 11-23

Understanding VLANs

A VLAN is a switched network that is logically segmented by function, project team, or application, without regard to the physical locations of the users. VLANs have the same attributes as physical LANs, but you can group end stations even if they are not physically located on the same LAN segment. Any switch port can belong to a VLAN, and unicast, broadcast, and multicast packets are forwarded and flooded only to end stations in the VLAN. Each VLAN is considered a logical network, and packets destined for stations that do not belong to the VLAN must be forwarded through a router, as shown in Figure 11-1. Because a VLAN is considered a separate logical network, it contains its own bridge MIB information and can support its own implementation of spanning tree. See Chapter 14, “Configuring STP.”

Figure 11-1 shows an example of VLANs segmented into logically defined networks.

Figure 11-1 VLANs as Logically Defined Networks



VLANs are often associated with IP subnetworks. For example, all the end stations in a particular IP subnet belong to the same VLAN. Interface VLAN membership on the switch is assigned manually on an interface-by-interface basis. When you assign switch interfaces to VLANs by using this method, it is known as interface-based, or static, VLAN membership.

Note The switch does not support VLAN Trunking Protocol (VTP).

Traffic between VLANs must be routed. Switches that are running the metro IP access image can route traffic between VLANs by using switch virtual interfaces (SVIs). To route traffic between VLANs, an SVI must be explicitly configured and assigned an IP address. For more information, see the "Switch Virtual Interfaces" section on page 9-5 and the "Configuring Layer 3 Interfaces" section on page 9-20.

This section includes these topics:

- Supported VLANs, page 11-3
- Normal-Range VLANs, page 11-3
- Extended-Range VLANs, page 11-4
- VLAN Port Membership Modes, page 11-4
- UNI VLANs, page 11-5

Supported VLANs

VLANs are identified with a number from 1 to 4094. VLAN IDs 1002 through 1005 are reserved for Token Ring and FDDI VLANs. VLAN IDs greater than 1005 are extended-range VLANs and are not stored in the VLAN database.

Although the switch supports a total of 1005 (normal-range and extended-range) VLANs, the number of routed ports, SVIs, and other configured features affects the use of the switch hardware.

The switch supports per-VLAN spanning-tree plus (PVST+) or rapid PVST+ with a maximum of 128 spanning-tree instances. One spanning-tree instance is allowed per VLAN.

Note Only network node interfaces (NNIs) support STP. User network interfaces (UNIs) by default are always in a forwarding state.

See the "VLAN Configuration Guidelines" section on page 11-8 for more information about the number of spanning-tree instances and the number of VLANs. The switch supports IEEE 802.1Q trunking for sending VLAN traffic over Ethernet ports.

Normal-Range VLANs

Normal-range VLANs are VLANs with VLAN IDs 1 to 1005. You can add, modify or remove configurations for VLANs 2 to 1001 in the VLAN database. (VLAN IDs 1 and 1002 to 1005 are automatically created and cannot be removed.)

Configurations for VLAN IDs 1 to 1005 are written to the file *vlan.dat* (VLAN database), and you can display them by entering the **show vlan** privileged EXEC command. The *vlan.dat* file is stored in flash memory.

Caution You can cause inconsistency in the VLAN database if you try to manually delete the *vlan.dat* file. If you want to modify the VLAN configuration, use the commands described in these sections and in the command reference for this release.

You can set these parameters when you create a new normal-range VLAN or modify an existing VLAN in the VLAN database:

- VLAN ID
- VLAN name
- VLAN type (Ethernet, Fiber Distributed Data Interface [FDDI], FDDI network entity title [NET], TrBRF, or TrCRF, Token Ring, Token Ring-Net)

Note The switch supports only Ethernet VLANs. You can configure parameters for FDDI and Token Ring VLANs and view the results in the vlan.dat file, but these parameters are not used.

- VLAN state (active or suspended)
- Maximum transmission unit (MTU) for the VLAN
- Security Association Identifier (SAID)
- Bridge identification number for TrBRF VLANs

- Ring number for FDDI and TrCRF VLANs
- Parent VLAN number for TrCRF VLANs
- Spanning Tree Protocol (STP) type for TrCRF VLANs
- VLAN number to use when translating from one VLAN type to another
- Private VLAN. Configure the VLAN as a primary or secondary private VLAN. For information about private VLANs, see Chapter 12, "Configuring Private VLANs."
- Remote SPAN VLAN. Configure the VLAN as the Remote Switched Port Analyzer (RSPAN) VLAN for a remote SPAN session. For more information on remote SPAN, see Chapter 24, "Configuring SPAN and RSPAN."
- UNI VLAN configuration

For extended-range VLANs, you can configure only MTU, private VLAN, remote SPAN VLAN, and UNI VLAN parameters.

Note This chapter does not provide configuration details for most of these parameters. For complete information on the commands and parameters that control VLAN configuration, see the command reference for this release.

Extended-Range VLANs

You can create extended-range VLANs (in the range 1006 to 4094) to enable service providers to extend their infrastructure to a greater number of customers. The extended-range VLAN IDs are allowed for any **switchport** commands that allow VLAN IDs. Extended-range VLAN configurations are not stored in the VLAN database, but they are stored in the switch running configuration file, and you can save the configuration in the startup configuration file by using the **copy running-config startup-config** privileged EXEC command.

Note Although the switch supports 4094 VLAN IDs, the actual number of VLANs supported is 1005.

VLAN Port Membership Modes

You configure a port to belong to a VLAN by assigning a membership mode that specifies the kind of traffic that the port carries and the number of VLANs to which it can belong. Table 11-1 lists the membership modes and characteristics.

Table 11-1 Port Membership Modes

Membership Mode	VLAN Membership Characteristics
Static-access	A static-access port can belong to one VLAN and is manually assigned to that VLAN. For more information, see the "Assigning Static-Access Ports to a VLAN" section on page 11-10.
Trunk (IEEE 802.1Q)	A trunk port is a member of all VLANs by default, including extended-range VLANs, but membership can be limited by configuring the allowed-VLAN list. For information about configuring trunk ports, see the "Configuring an Ethernet Interface as a Trunk Port" section on page 11-16.

Table 11-1 Port Membership Modes (continued)

Membership Mode	VLAN Membership Characteristics
Dynamic-access	A dynamic-access port can belong to one VLAN (VLAN ID 1 to 4094) and is dynamically assigned by a VMPS. The VMPS can be a Catalyst 5000 or Catalyst 6500 series switch, for example, but never a Cisco ME 3400 Ethernet Access switch. The Cisco ME 3400 switch is a VMPS client. **Note** Only UNIs can be dynamic-access ports. You can have dynamic-access ports and trunk ports on the same switch, but you must connect the dynamic-access port to an end station or hub and not to another switch. For configuration information, see the "Configuring Dynamic-Access Ports on VMPS Clients" section on page 11-26.
Private VLAN	A private VLAN port is a host or promiscuous port that belongs to a private VLAN primary or secondary VLAN. Only NNIs can be configured as promiscuous ports. For information about private VLANs, see Chapter 12, "Configuring Private VLANs."
Tunnel (**dot1q-tunnel**)	Tunnel ports are used for IEEE 802.1Q tunneling to maintain customer VLAN integrity across a service-provider network. You configure a tunnel port on an edge switch in the service-provider network and connect it to an IEEE 802.1Q trunk port on a customer interface, creating an assymetric link. A tunnel port belongs to a single VLAN that is dedicated to tunneling. Tunneling is supported only when the switch is running the metro access or metro IP access image. For more information about tunnel ports, see Chapter 13, "Configuring IEEE 802.1Q and Layer 2 Protocol Tunneling."

For more detailed definitions of access and trunk modes and their functions, see Table 11-4 on page 11-15.

When a port belongs to a VLAN, the switch learns and manages the addresses associated with the port on a per-VLAN basis. For more information, see the "Managing the MAC Address Table" section on page 5-19.

UNI VLANs

The Cisco ME switch is the boundary between customer networks and the service-provider network, with user network interfaces (UNIs) connected to the customer side of the network. When customer traffic enters or leaves the service-provider network, the customer VLAN ID must be isolated from other customers' VLAN IDs. You can achieve this isolation by several methods, including using private VLANs. On the Cisco ME switch, this isolation occurs by default by using UNI VLANs.

There are two types of UNI VLANs:

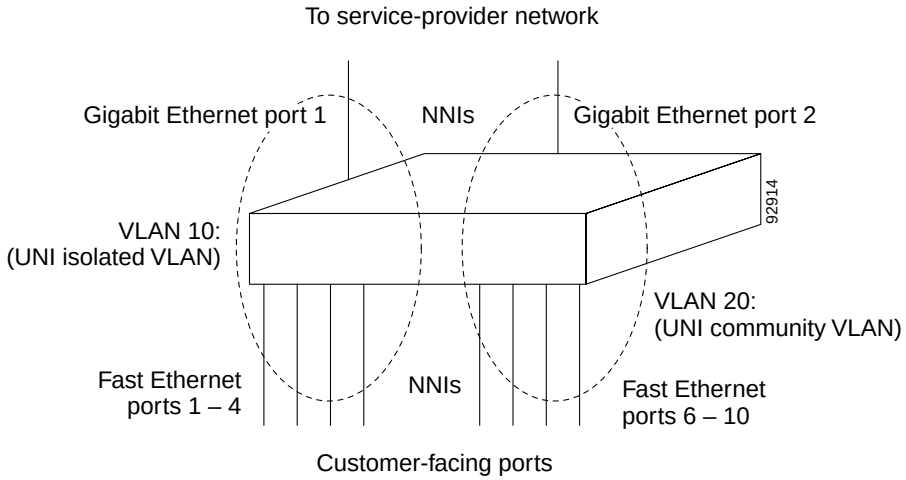
- UNI isolated VLAN—This is the default VLAN state for all VLANs created on the switch. Local switching does not occur among UNIs on the switch that belong to the same UNI isolated VLAN. This configuration is designed for cases when different customers are connected to UNIs on the same switch. However, switching is allowed among UNIs on different switches even though they belong to the same UNI isolated VLAN.

- UNI community VLAN—Local switching is allowed among UNIs on the switch that belong to the same UNI community VLAN. If UNIs belong to the same customer, and you want to switch packets between the ports, you can configure the common VLAN as a UNI community VLAN. There is no local switching between ports in a UNI community VLAN and ports outside of the VLAN. The switch does not support more than eight UNIs in a UNI community VLAN.

Network node interfaces (NNIs) are not affected by the type of UNI VLAN to which they belong. Switching can occur between NNIs and other NNIs or UNIs on the switch or other switches that are part of the same VLAN, regardless of VLAN type.

In the configuration in Figure 11-2, if VLAN 10 is a UNI isolated VLAN and VLAN 20 is a UNI community VLAN, local switching does not take place among Fast Ethernet ports 1-4, but local switching can occur between Fast Ethernet ports 6-10. The NNIs in both VLAN 10 and VLAN 20 can exchange packets with the UNIs in the same VLAN.

Figure 11-2 UNI Isolated and Community VLANs in the Cisco ME Switch



A UNI can be an access port, a trunk port, a private VLAN port, or an IEEE 802.1Q tunnel port. It can also be a member of an EtherChannel.

When a UNI configured as an IEEE 802.1Q trunk port belongs to a UNI isolated VLAN, the VLAN on the trunk is isolated from the same VLAN ID on a different trunk port or an access port. Other VLANs on the trunk port can be of different types (private VLAN, UNI community VLAN, and so on). For example, a UNI access port and one VLAN on a UNI trunk port can belong to the same UNI isolated VLAN. In this case, isolation occurs between the UNI access port and the VLAN on the UNI trunk port. Other access ports and other VLANs on the trunk port are isolated because they belong to different VLANs.

UNIs and NNIs are always isolated from ports on different VLANs.

Creating and Modifying VLANs

You use VLAN configuration mode, accessed by entering the **vlan** global configuration command to create VLANs and to modify some parameters. You use the interface configuration mode to define the port membership mode and to add and remove ports from VLANs. The results of these commands are written to the running-configuration file, and you can display the file by entering the **show running-config** privileged EXEC command.

These sections contain VLAN configuration information:

- Default Ethernet VLAN Configuration, page 11-7
- VLAN Configuration Guidelines, page 11-8
- Creating or Modifying an Ethernet VLAN, page 11-9
- Assigning Static-Access Ports to a VLAN, page 11-10
- Creating an Extended-Range VLAN with an Internal VLAN ID, page 11-11
- Configuring UNI VLANs, page 11-12

If the switch is running the metro IP access or metro access image, for more efficient management of the MAC address table space available on the switch, you can control which VLANs learn MAC addresses by disabling MAC address learning on specific VLANs. See the "Disabling MAC Address Learning on a VLAN" section on page 5-26 for more information.

Default Ethernet VLAN Configuration

The switch supports only Ethernet interfaces. Table 11-2 shows the default configuration for Ethernet VLANs.

Note On extended-range VLANs, you can change only the MTU size, the private VLAN, the remote SPAN, and the UNI VLAN configuration. All other characteristics must remain at the default conditions.

Table 11-2 Ethernet VLAN Defaults and Ranges

Parameter	Default	Range
VLAN ID	1	1 to 4094. **Note** Extended-range VLANs (VLAN IDs 1006 to 4094) are not saved in the VLAN database.
VLAN name	*VLANxxxx*, where *xxxx* represents four numeric digits (including leading zeros) equal to the VLAN ID number	No range
IEEE 802.10 SAID	100001 (100000 plus the VLAN ID)	1 to 4294967294
MTU size	1500	1500 to 9198
Translational bridge 1	0	0 to1005
Translational bridge 2	0	0 to1005
VLAN state	active	active, suspend
Remote SPAN	disabled	enabled, disabled
Private VLANs	none configured	2 to 1001, 1006 to 4094.
UNI VLAN	UNI isolated VLAN	2 to 1001, 1006 to 4094. VLAN 1 is always a UNI isolated VLAN.

VLAN Configuration Guidelines

Follow these guidelines when creating and modifying VLANs in your network:

- The switch supports 1005 VLANs.
- Normal-range Ethernet VLANs are identified with a number between 1 and 1001. VLAN numbers 1002 through 1005 are reserved for Token Ring and FDDI VLANs.
- The switch does not support Token Ring or FDDI media. The switch does not forward FDDI, FDDI-Net, TrCRF, or TrBRF traffic.
- VLAN configurations for VLANs 1 to 1005 are always saved in the VLAN database and in the switch running configuration file.
- Configuration options for VLAN IDs 1006 through 4094 (extended-range VLANs) are limited to MTU, RSPAN VLAN, private VLAN, and UNI VLAN. Extended-range VLANs are not saved in the VLAN database.
- Spanning Tree Protocol (STP) is enabled by default for only NNIs on all VLANs. The switch supports 128 spanning-tree instances. If a switch has more active VLANs than supported spanning-tree instances, spanning tree can be enabled on 128 VLANs and is disabled on the remaining VLANs. If you have already used all available spanning-tree instances on a switch, adding another VLAN creates a VLAN on that switch that is not running spanning tree. If you have the default allowed list on the trunk ports of that switch (which is to allow all VLANs), the new VLAN is carried on all trunk ports. Depending on the topology of the network, this could create a loop in the new VLAN that would not be broken, particularly if there are several adjacent switches that all have run out of spanning-tree instances. You can prevent this possibility by setting allowed lists on the trunk ports of switches that have used up their allocation of spanning-tree instances.

 If the number of VLANs on the switch exceeds the number of supported spanning-tree instances, we recommend that you configure the IEEE 802.1s Multiple STP (MSTP) on your switch to map multiple VLANs to a single spanning-tree instance. For more information about MSTP, see Chapter 15, “Configuring MSTP.”

 Note MSTP is supported only on NNIs.

- Each routed port on the switch creates an internal VLAN for its use. These internal VLANs use extended-range VLAN numbers, and the internal VLAN ID cannot be used for an extended-range VLAN. If you try to create an extended-range VLAN with a VLAN ID that is already allocated as an internal VLAN, an error message is generated, and the command is rejected.
 - Because internal VLAN IDs are in the lower part of the extended range, we recommend that you create extended-range VLANs beginning from the highest number (4094) and moving to the lowest (1006) to reduce the possibility of using an internal VLAN ID.
 - Before configuring extended-range VLANs, enter the **show vlan internal usage** privileged EXEC command to see which VLANs have been allocated as internal VLANs.
 - If necessary, you can shut down the routed port assigned to the internal VLAN, which frees up the internal VLAN, and then create the extended-range VLAN and re-enable the port, which then uses another VLAN as its internal VLAN. See the “Creating an Extended-Range VLAN with an Internal VLAN ID” section on page 11-11.
- Although the switch supports a total of 1005 (normal-range and extended-range) VLANs, the number of routed ports, SVIs, and other configured features affects the use of the switch hardware. If you try to create an extended-range VLAN and there are not enough hardware resources available, an error message is generated, and the extended-range VLAN is rejected.

Creating or Modifying an Ethernet VLAN

To access VLAN configuration mode, enter the **vlan** global configuration command with a VLAN ID. Enter a new VLAN ID to create a VLAN, or enter an existing VLAN ID to modify that VLAN. You can use the default VLAN configuration (Table 11-2) or enter commands to configure the VLAN.

Note Extended-range VLANs use the default Ethernet VLAN characteristics and the MTU, the private VLAN, the RSPAN, and the UNI-VLAN configurations are the only parameters you can change.

For more information about commands available in this mode, see the **vlan** command description in the command reference for this release. When you have finished the configuration, you must exit VLAN configuration mode for the configuration to take effect. To display the VLAN configuration, enter the **show vlan** privileged EXEC command.

The configurations of VLAN IDs 1 to 1005 are always saved in the VLAN database (vlan.dat file) with a VLAN number and name and in the switch running configuration file. Extended-range VLANs are not saved in the VLAN database; they are saved in the switch running configuration file. You can save the VLAN configuration in the switch startup configuration file by using the **copy running-config startup-config** privileged EXEC command.

Note Before you create an extended-range VLAN, you can verify that the VLAN ID is not used internally by entering the **show vlan internal usage** privileged EXEC command. If the VLAN ID is used internally and you want to release it, go to the "Creating an Extended-Range VLAN with an Internal VLAN ID" section on page 11-11 before creating the extended-range VLAN.

Beginning in privileged EXEC mode, follow these steps to create or modify an Ethernet VLAN:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**vlan** *vlan-id*	Enter a VLAN ID, and enter VLAN configuration mode. Enter a new VLAN ID to create a VLAN, or enter an existing VLAN ID to modify that VLAN. The available VLAN ID range for this command is 1 to 4094. **Note** When you create a new VLAN, by default the VLAN is a UNI isolated VLAN.
Step 3	**name** *vlan-name*	(Optional and supported on normal-range VLANs only) Enter a name for the VLAN. If no name is entered for the VLAN, the default in the VLAN database is to append the *vlan-id* with leading zeros to the word VLAN. For example, VLAN0004 is a default VLAN name for VLAN 4.
Step 4	**mtu** *mtu-size*	(Optional) Change the MTU size.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show vlan** {**name** *vlan-name* \| **id** *vlan-id*}	Verify your entries. The **name** option is only valid for VLAN IDs 1 to 1005.
Step 7	**copy running-config startup config**	(Optional) Save the configuration in the switch startup configuration file.

To delete a VLAN, use the **no vlan** *vlan-id* global configuration command. You cannot delete VLAN 1 or VLANs 1002 to 1005.

Caution When you delete a VLAN, any ports assigned to that VLAN become inactive. They remain associated with the VLAN (and thus inactive) until you assign them to a new VLAN.

To return the VLAN name to the default settings, use the **no name** or **no mtu** VLAN configuration command.

This example shows how to create Ethernet VLAN 20, name it *test20*, and add it to the VLAN database:

```
Switch# configure terminal
Switch(config)# vlan 20
Switch(config-vlan)# name test20
Switch(config-vlan)# end
```

This example shows how to create a new extended-range VLAN with all default characteristics, enter config-vlan mode, and save the new VLAN in the switch startup configuration file:

```
Switch(config)# vlan 2000
Switch(config-vlan)# end
Switch# copy running-config startup config
```

Assigning Static-Access Ports to a VLAN

You can assign a static-access port to a VLAN.

Note If you assign an interface to a VLAN that does not exist, the new VLAN is created. (See the "Creating or Modifying an Ethernet VLAN" section on page 11-9.)

Beginning in privileged EXEC mode, follow these steps to assign a port to a VLAN in the VLAN database:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode
Step 2	**interface** *interface-id*	Enter the interface to be added to the VLAN.
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled, and NNIs are enabled.
Step 4	**switchport mode access**	Define the VLAN membership mode for the port (Layer 2 access port).
Step 5	**switchport access vlan** *vlan-id*	Assign the port to a VLAN. Valid VLAN IDs are 1 to 4094.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show running-config interface** *interface-id*	Verify the VLAN membership mode of the interface.
Step 8	**show interfaces** *interface-id* **switchport**	Verify your entries in the *Administrative Mode* and the *Access Mode VLAN* fields of the display.
Step 9	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return an interface to its default configuration, use the **default interface** *interface-id* interface configuration command.

This example shows how to configure a port as an access port in VLAN 2:

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# interface fastethernet0/1
Switch(config-if)# switchport mode access
Switch(config-if)# switchport access vlan 2
Switch(config-if)# end
```

Creating an Extended-Range VLAN with an Internal VLAN ID

If you enter an extended-range VLAN ID that is already assigned to an internal VLAN, an error message appears, and the extended-range VLAN is rejected. To manually release an internal VLAN ID, you must temporarily shut down the routed port that is using the internal VLAN ID.

Beginning in privileged EXEC mode, follow these steps to release a VLAN ID that is assigned to an internal VLAN and to create an extended-range VLAN with that ID:

	Command	Purpose
Step 1	**show vlan internal usage**	Display the VLAN IDs being used internally by the switch. If the VLAN ID that you want to use is an internal VLAN, the display shows the routed port that is using the VLAN ID. Enter that port number in Step 3.
Step 2	**configure terminal**	Enter global configuration mode.
Step 3	**interface** *interface-id*	Specify the interface ID for the routed port that is using the VLAN ID, and enter interface configuration mode.
Step 4	**shutdown**	Shut down the port to release the internal VLAN ID.
Step 5	**exit**	Return to global configuration mode.
Step 6	**vlan** *vlan-id*	Enter the new extended-range VLAN ID, and enter config-vlan mode.
Step 7	**exit**	Exit from config-vlan mode, and return to global configuration mode.
Step 8	**interface** *interface-id*	Specify the interface ID for the routed port that you shut down in Step 4, and enter interface configuration mode.
Step 9	**no shutdown**	Re-enable the routed port. It will be assigned a new internal VLAN ID.
Step 10	**end**	Return to privileged EXEC mode.
Step 11	**copy running-config startup config**	(Optional) Save your entries in the switch startup configuration file.

Configuring UNI VLANs

By default, every VLAN configured on the switch is a UNI isolated VLAN. You can change VLAN configuration to that of a UNI community VLAN, a private VLAN. or an RSPAN VLAN. You can also change the configuration of one of these VLANs to the default of a UNI isolated VLAN.

Configuration Guidelines

These are guidelines for UNI VLAN configuration:

- UNI isolated VLANs have no effect on NNI ports.
- A UNI community VLAN is like a traditional VLAN except that it can include no more than eight UNIs.

To change a VLAN type, first enter the **vlan** *vlan-id* global configuration command to enter VLAN configuration mode:

 - To change a VLAN from UNI isolated VLAN to a private VLAN, enter the **private-vlan** VLAN configuration command.
 - To change a UNI community VLAN to a private VLAN, you must first remove the community VLAN type by entering the **no uni-vlan** VLAN configuration command. Then enter the **private-vlan** VLAN configuration command.
 - To change a VLAN from a UNI isolated VLAN to an RSPAN VLAN, enter the **rspan-vlan** VLAN configuration command.
 - To change a UNI community VLAN to an RSPAN VLAN, you must first remove the community VLAN type by entering the **no uni-vlan** VLAN configuration command. Then enter the **rspan-vlan** VLAN configuration command.
 - To change a private VLAN to a UNI VLAN, you must first remove the private VLAN type by entering the **no private-vlan** VLAN configuration command. Then enter the **uni-vlan** VLAN configuration command.
 - To change an RSPAN VLAN to a UNI VLAN, you must first remove the RSPAN VLAN type by entering the **no rspan-vlan** VLAN configuration command. Then enter the **uni-vlan** VLAN configuration command

- The switch does not support more than eight UNIs in a community VLAN. You cannot configure a VLAN as a UNI community VLAN if more than eight UNIs belong to the VLAN.
- If you attempt to add a UNI static access port to a UNI community VLAN that already has eight UNIs, the configuration is refused. If a UNI dynamic access port is added to a UNI community VLAN that already has eight UNIs, the port is error-disabled.

Configuring UNI VLANs

By default, every VLAN created on the switch is a UNI isolated VLAN. You can change the configuration to UNI community VLAN or to a private VLAN or RSPAN VLAN. For procedures for configuring private VLANs or RSPAN VLANs, see Chapter 12, "Configuring Private VLANs" and Chapter 24, "Configuring SPAN and RSPAN."

Beginning in privileged EXEC mode, follow these steps to change the type of a UNI VLAN:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**vlan** *vlan-id*	Enter a VLAN ID, and enter VLAN configuration mode. Enter a new VLAN ID to create a VLAN, or enter an existing VLAN ID to modify that VLAN. By default, the VLAN is a UNI isolated VLAN. **Note** The available VLAN ID range for this command is 1 to 4094.
Step 3	**uni-vlan** {**community** \| **isolated**}	Configure the UNI VLAN type. • Enter **community** to change from the default to a UNI community VLAN. • Enter **isolated** to return to the default UNI isolated VLAN. **Note** VLAN 1 is always a UNI isolated VLAN; you cannot configure VLAN 1 as a UNI community VLAN. The reserved VLANs 1002 to 1005 are not Ethernet VLANs.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show vlan uni-vlan** [**type**]	Display UNI VLAN information. Enter **type** (optional) to see only the VLAN ID and type of UNI VLAN.
Step 6	**copy running-config startup config**	(Optional) Save the configuration in the switch startup configuration file.

Use the **no uni-vlan** VLAN configuration command to return to the default (UNI isolated VLAN). Entering **uni-vlan isolated** command has the same effect as entering the no **uni-vlan** VLAN configuration command. The **show vlan** and **show vlan** *vlan-id* privileged EXEC commands also display UNI VLAN information, but only UNI community VLANs appear. To display both isolated and community VLANs, use the **show vlan uni-vlan type** command.

Displaying VLANs

Use the **show vlan** privileged EXEC command to display a list of all VLANs on the switch, including extended-range VLANs. The display includes VLAN status, ports, and configuration information. Table 11-3 lists other privileged EXEC commands for monitoring VLANs.

Table 11-3 VLAN Monitoring Commands

Command	Purpose
show interfaces [**vlan** *vlan-id*]	Display characteristics for all interfaces or for the specified VLAN configured on the switch.
show vlan [**id** *vlan-id*]	Display parameters for all VLANs or the specified VLAN on the switch.
show vlan [*vlan-name*] **uni-vlan type**	Display UNI isolated or UNI community VLANs by VLAN name.
show vlan uni-vlan	Display UNI community VLANs and associated ports on the switch.
show vlan uni-vlan type	Display UNI isolated and UNI community VLANs on the switch by VLAN ID.

For more details about the **show** command options and explanations of output fields, see the command reference for this release.

Configuring VLAN Trunks

These sections contain this conceptual information:

- Trunking Overview, page 11-14
- Default Layer 2 Ethernet Interface VLAN Configuration, page 11-16
- Configuring an Ethernet Interface as a Trunk Port, page 11-16
- Configuring Trunk Ports for Load Sharing, page 11-19

Trunking Overview

A trunk is a point-to-point link between one or more Ethernet switch interfaces and another networking device such as a router or a switch. Ethernet trunks carry the traffic of multiple VLANs over a single link, and you can extend the VLANs across an entire network. The switch supports the IEEE 802.1Q industry-standard trunking encapsulation.

You can configure a trunk on a single Ethernet interface or on an EtherChannel bundle. For more information about EtherChannels, see Chapter 31, “Configuring EtherChannels.”

Ethernet interfaces support different trunking modes (see Table 11-4). You can set an interface as trunking or nontrunking.

- If you do not intend to trunk across links, use the **switchport mode access** interface configuration command to disable trunking.

- To enable trunking, use the **switchport mode trunk** interface configuration command to change the interface to a trunk.

Table 11-4 Layer 2 Interface Modes

Mode	Function
switchport mode access	Puts the interface (access port) into permanent nontrunking mode and negotiates to convert the link into a nontrunk link. The interface becomes a nontrunk interface regardless of whether or not the neighboring interface is a trunk interface. This is the default mode.
switchport mode trunk	Puts the interface into permanent trunking mode and negotiates to convert the neighboring link into a trunk link. The interface becomes a trunk interface even if the neighboring interface is not a trunk interface.
switchport mode dot1q-tunnel	Configures the interface as a tunnel (nontrunking) port to be connected in an asymmetric link with an IEEE 802.1Q trunk port. The IEEE 802.1Q tunneling is used to maintain customer VLAN integrity across a service provider network. See Chapter 13, "Configuring IEEE 802.1Q and Layer 2 Protocol Tunneling," for more information on tunnel ports.
switchport mode private-vlan	Configure the interface as a private VLAN host or promiscuous port (only NNIs can be configured as promiscuous ports). For information about private VLANs, see Chapter 12, "Configuring Private VLANs."

IEEE 802.1Q Configuration Considerations

The IEEE 802.1Q trunks impose these limitations on the trunking strategy for a network:

- In a network of Cisco switches connected through IEEE 802.1Q trunks, the switches maintain one spanning-tree instance for each VLAN allowed on the trunks. Non-Cisco devices might support one spanning-tree instance for all VLANs.

 When you connect a Cisco switch to a non-Cisco device through an IEEE 802.1Q trunk, the Cisco switch combines the spanning-tree instance of the VLAN of the trunk with the spanning-tree instance of the non-Cisco IEEE 802.1Q switch. However, spanning-tree information for each VLAN is maintained by Cisco switches separated by a cloud of non-Cisco IEEE 802.1Q switches. The non-Cisco IEEE 802.1Q cloud separating the Cisco switches is treated as a single trunk link between the switches.

- Make sure that the native VLAN for an IEEE 802.1Q trunk is the same on both ends of the trunk link. If the native VLAN on one end of the trunk is different from the native VLAN on the other end, spanning-tree loops might result.

- Disabling spanning tree on the native VLAN of an IEEE 802.1Q trunk without disabling spanning tree on every VLAN in the network can potentially cause spanning-tree loops. We recommend that you leave spanning tree enabled on the native VLAN of an IEEE 802.1Q trunk or disable spanning tree on every VLAN in the network. Make sure that your network is loop-free before disabling spanning tree.

Default Layer 2 Ethernet Interface VLAN Configuration

Table 11-5 shows the default Layer 2 Ethernet interface VLAN configuration.

Table 11-5 Default Layer 2 Ethernet Interface VLAN Configuration

Feature	Default Setting
Interface mode	**switchport mode access**
Allowed VLAN range	VLANs 1 to 4094
Default VLAN (for access ports)	VLAN 1
Native VLAN (for IEEE 802.1Q trunks)	VLAN 1

Configuring an Ethernet Interface as a Trunk Port

These sections contain this configuration information about trunk ports:

- Interaction with Other Features, page 11-16
- Defining the Allowed VLANs on a Trunk, page 11-17
- Configuring the Native VLAN for Untagged Traffic, page 11-19
- Configuring the Native VLAN for Untagged Traffic, page 11-19

Interaction with Other Features

Trunking interacts with other features in these ways:

- A trunk port cannot be a secure port.
- A trunk port cannot be a tunnel port.
- Trunk ports can be grouped into EtherChannel port groups, but all trunks in the group must have the same configuration. When a group is first created, all ports follow the parameters set for the first port to be added to the group. If you change the configuration of one of these parameters, the switch propagates the setting that you entered to all ports in the group:
 - allowed-VLAN list.
 - STP port priority for each VLAN (NNIs only).
 - STP Port Fast setting (NNIs only).
 - trunk status: if one port in a port group ceases to be a trunk, all ports cease to be trunks.
- If you try to enable IEEE 802.1x on a trunk port, an error message appears, and IEEE 802.1x is not enabled. If you try to change the mode of an IEEE 802.1x-enabled port to trunk, the port mode is not changed.

Configuring a Trunk Port

Beginning in privileged EXEC mode, follow these steps to configure a port as an IEEE 802.1Q trunk port:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the port to be configured for trunking, and enter interface configuration mode.
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled, and NNIs are enabled.
Step 4	**switchport mode trunk**	Configure the interface as a Layer 2 trunk.
Step 5	**switchport access vlan** *vlan-id*	(Optional) Specify the default VLAN, which is used if the interface stops trunking.
Step 6	**switchport trunk native vlan** *vlan-id*	Specify the native VLAN for IEEE 802.1Q trunks.
Step 7	**end**	Return to privileged EXEC mode.
Step 8	**show interfaces** *interface-id* **switchport**	Display the switchport configuration of the interface in the *Administrative Mode* field of the display.
Step 9	**show interfaces** *interface-id* **trunk**	Display the trunk configuration of the interface.
Step 10	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return an interface to its default configuration, use the **default interface** *interface-id* interface configuration command. To reset all trunking characteristics of a trunking interface to the defaults, use the **no switchport trunk** interface configuration command. To disable trunking, use the **switchport mode access** interface configuration command to configure the port as a static-access port.

This example shows how to configure a port as an IEEE 802.1Q trunk with VLAN 33 as the native VLAN:

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# interface fastethernet0/2
Switch(config-if)# switchport mode trunk
Switch(config-if)# switchport trunk native vlan 33
Switch(config-if)# end
```

Defining the Allowed VLANs on a Trunk

By default, a trunk port sends traffic to and receives traffic from all VLANs. All VLAN IDs, 1 to 4094, are allowed on each trunk. However, you can remove VLANs from the allowed list, preventing traffic from those VLANs from passing over the trunk. To restrict the traffic a trunk carries, use the **switchport trunk allowed vlan remove** *vlan-list* interface configuration command to remove specific VLANs from the allowed list.

Note VLAN 1 is the default VLAN on all trunk ports in all Cisco switches, and it has previously been a requirement that VLAN 1 always be enabled on every trunk link. The VLAN 1 minimization feature allows you to disable VLAN 1 on any individual VLAN trunk link so that no user traffic (including spanning-tree advertisements) is sent or received on VLAN 1. You do this by removing VLAN 1 from the allowed VLAN list.

To reduce the risk of spanning-tree loops or storms, you can disable VLAN 1 on any individual VLAN trunk port by removing VLAN 1 from the allowed list. When you remove VLAN 1 from a trunk port, the interface continues to send and receive management traffic, for example, Cisco Discovery Protocol (CDP), Port Aggregation Protocol (PAgP), and Link Aggregation Control Protocol (LACP) in VLAN 1.

If a trunk port with VLAN 1 disabled is converted to a nontrunk port, it is added to the access VLAN. If the access VLAN is set to 1, the port is added to VLAN 1, regardless of the **switchport trunk allowed** setting. The same is true for any VLAN that has been disabled on the port.

A trunk port can become a member of a VLAN if the VLAN is enabled and if the VLAN is in the allowed list for the port.

Beginning in privileged EXEC mode, follow these steps to modify the allowed list of an IEEE 802.1Q trunk:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the port to be configured, and enter interface configuration mode.
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled, and NNIs are enabled.
Step 4	**switchport mode trunk**	Configure the interface as a VLAN trunk port.
Step 5	**switchport trunk allowed vlan** {**add** \| **all** \| **except** \| **remove**} *vlan-list*	(Optional) Configure the list of VLANs allowed on the trunk. For explanations about using the **add**, **all**, **except**, and **remove** keywords, see the command reference for this release. The *vlan-list* parameter is either a single VLAN number from 1 to 4094 or a range of VLANs described by two VLAN numbers, the lower one first, separated by a hyphen. Do not enter any spaces between comma-separated VLAN parameters or in hyphen-specified ranges. All VLANs are allowed by default.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show interfaces** *interface-id* **switchport**	Verify your entries in the *Trunking VLANs Enabled* field of the display.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default allowed VLAN list of all VLANs, use the **no switchport trunk allowed vlan** interface configuration command.

This example shows how to remove VLAN 2 from the allowed VLAN list on a port:

```
Switch(config)# interface fastethernet0/1
Switch(config-if)# switchport trunk allowed vlan remove 2
Switch(config-if)# end
```

Configuring the Native VLAN for Untagged Traffic

A trunk port configured with IEEE 802.1Q tagging can receive both tagged and untagged traffic. By default, the switch forwards untagged traffic in the native VLAN configured for the port. The native VLAN is VLAN 1 by default.

Note The native VLAN can be assigned any VLAN ID.

For information about IEEE 802.1Q configuration issues, see the "IEEE 802.1Q Configuration Considerations" section on page 11-15.

Beginning in privileged EXEC mode, follow these steps to configure the native VLAN on an IEEE 802.1Q trunk:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Define the interface that is configured as the IEEE 802.1Q trunk, and enter interface configuration mode.
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled and NNIs are enabled.
Step 4	**switchport trunk native vlan** *vlan-id*	Configure the VLAN that is sending and receiving untagged traffic on the trunk port. For *vlan-id*, the range is 1 to 4094.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show interfaces** *interface-id* **switchport**	Verify your entries in the *Trunking Native Mode VLAN* field.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default native VLAN, VLAN 1, use the **no switchport trunk native vlan** interface configuration command.

If a packet has a VLAN ID that is the same as the sending port native VLAN ID, the packet is sent untagged; otherwise, the switch sends the packet with a tag.

Configuring Trunk Ports for Load Sharing

Load sharing divides the bandwidth supplied by parallel trunks that connect switches. To avoid loops, STP normally blocks all but one parallel link between switches. Using load sharing, you divide the traffic between the links according to the VLAN to which the traffic belongs.

You configure load sharing on NNI trunk ports by using STP port priorities or STP path costs. For load sharing using STP port priorities, both load-sharing links must be connected to the same switch. For load sharing using STP path costs, each load-sharing link can be connected to the same switch or to two different switches. For more information about STP, see Chapter 14, "Configuring STP."

Load Sharing Using STP Port Priorities

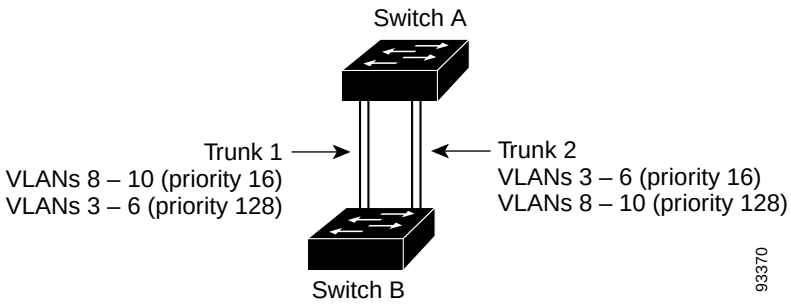
When two ports on the same switch form a loop, the switch uses the STP port priority to decide which port is enabled and which port is in a blocking state. You can set the priorities on a parallel NNI trunk port so that the port carries all the traffic for a given VLAN. The NNI trunk port with the higher priority (lower values) for a VLAN is forwarding traffic for that VLAN. The NNI trunk port with the lower priority (higher values) for the same VLAN remains in a blocking state for that VLAN. One trunk port sends or receives all traffic for the VLAN.

Figure 11-3 shows two trunks connecting supported switches. In this example, the switches are configured as follows:

- VLANs 8 through 10 are assigned a port priority of 16 on Trunk 1.
- VLANs 3 through 6 retain the default port priority of 128 on Trunk 1.
- VLANs 3 through 6 are assigned a port priority of 16 on Trunk 2.
- VLANs 8 through 10 retain the default port priority of 128 on Trunk 2.

In this way, Trunk 1 carries traffic for VLANs 8 through 10, and Trunk 2 carries traffic for VLANs 3 through 6. If the active trunk fails, the trunk with the lower priority takes over and carries the traffic for all of the VLANs. No duplication of traffic occurs over any trunk port.

Figure 11-3 Load Sharing by Using STP Port Priorities



Beginning in privileged EXEC mode on Switch A, follow these steps to configure the network shown in Figure 11-3. Note that you can use any interface numbers; those shown are examples only.

	Command	Purpose
Step 8	**show vlan**	Verify that the referenced VLANs exist on Switch A. If not, create the VLANs by entering the VLAN IDs.
Step 9	**configure terminal**	Enter global configuration mode.
Step 10	**interface gigabitethernet 0/1**	Define the interface to be configured as the Trunk 1 interface, and enter interface configuration mode.
Step 11	**port-type nni**	Configure the interface as an NNI. UNIs do not support STP.
Step 12	**switchport mode trunk**	Configure the port as a trunk port.
Step 13	**spanning-tree vlan 8-10 port-priority 16**	Assign the port priority of 16 for VLANs 8 through 10 on Trunk 1.
Step 14	**end**	Return to privileged EXEC mode.
Step 15	**show interfaces gigabitethernet 0/1 switchport**	Verify the port configuration.
Step 16	**configure terminal**	Enter global configuration mode.

	Command	Purpose
Step 17	**interface gigabitethernet 0/2**	Define the interface to be configured as the Trunk 2 interface, and enter interface configuration mode.
Step 18	**port-type nni**	Configure the interface as an NNI.
Step 19	**switchport mode trunk**	Configure the port as a trunk port.
Step 20	**spanning-tree vlan 3-6 port-priority 16**	Assign the port priority of 16 for VLANs 3 through 6 on Trunk 2.
Step 21	**end**	Return to privileged EXEC mode.
Step 22	**show interfaces gigabitethernet 0/2 switchport**	Verify the port configuration.
Step 23	**show running-config**	Verify your entries.
Step 24	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Follow the same steps on Switch B to configure the trunk port for Trunk 1 with a spanning-tree port priority of 16 for VLANs 8 through 10, and the configure trunk port for Trunk 2 with a spanning-tree port priority of 16 for VLANs 3 through 6.

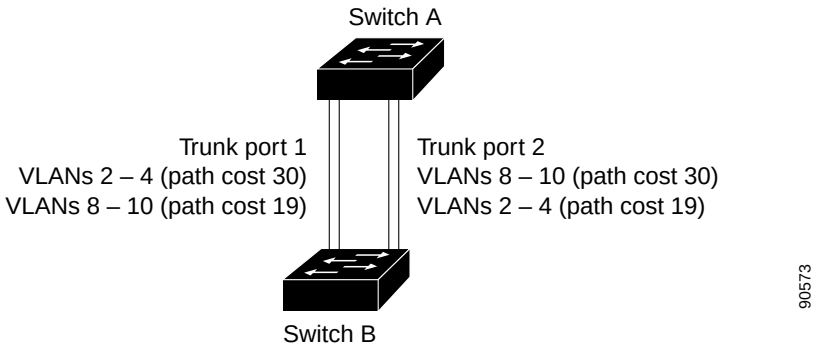
Load Sharing Using STP Path Cost

You can configure parallel trunks to share VLAN traffic by setting different path costs on a trunk and associating the path costs with different sets of VLANs, blocking different ports for different VLANs. The VLANs keep the traffic separate and maintain redundancy in the event of a lost link.

In Figure 11-4, Trunk ports 1 and 2 are configured as 100Base-T ports. These VLAN path costs are assigned:

- VLANs 2 through 4 are assigned a path cost of 30 on Trunk port 1.
- VLANs 8 through 10 retain the default 100Base-T path cost on Trunk port 1 of 19.
- VLANs 8 through 10 are assigned a path cost of 30 on Trunk port 2.
- VLANs 2 through 4 retain the default 100Base-T path cost on Trunk port 2 of 19.

Figure 11-4 Load-Sharing Trunks with Traffic Distributed by Path Cost



Beginning in privileged EXEC mode, follow these steps to configure the network shown in Figure 11-4:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode on Switch A.
Step 2	**interface fastethernet0/1**	Define the interface to be configured as Trunk port 1, and enter interface configuration mode.
Step 3	**port-type nni**	Configure the interface as an NNI.
Step 4	**switchport mode trunk**	Configure the port as a trunk port.
Step 5	**exit**	Return to global configuration mode.
Step 6	**interface fastethernet0/2**	Define the interface to be configured as Trunk port 2, and enter interface configuration mode.
Step 7	**port-type nni**	Configure the interface as an NNI.
Step 8	**switchport mode trunk**	Configure the port as a trunk port.
Step 9	**end**	Return to privileged EXEC mode.
Step 10	**show running-config**	Verify your entries. In the display, make sure that the interfaces configured in Steps 2 and 7 are configured as trunk ports.
Step 11	**show vlan**	Verify that VLANs 2 through 4 and 8 through 10 are configured on Switch A. If not, create these VLANs.
Step 12	**configure terminal**	Enter global configuration mode.
Step 13	**interface fastethernet0/1**	Enter interface configuration mode for Trunk port 2.
Step 14	**spanning-tree vlan 2-4 cost 30**	Set the spanning-tree path cost to 30 for VLANs 2 through 4.
Step 15	**exit**	Return to global configuration mode.
Step 16	**interface fastethernet0/2**	Enter interface configuration mode for Trunk port 2.
Step 17	**spanning-tree vlan 8-10 cost 30**	Set the spanning-tree path cost to 30 for VLANs 2 through 4.
Step 18	**exit**	Return to global configuration mode.
Step 19		Repeat Steps 9 through 11 on the other configured trunk interface on Switch A, and set the spanning-tree path cost to 30 for VLANs 8, 9, and 10.
Step 20	**exit**	Return to privileged EXEC mode.
Step 21	**show running-config**	Verify your entries. In the display, verify that the path costs are set correctly for both trunk interfaces.
Step 22	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Follow the same steps on Switch B to configure the trunk port for Trunk 1 with a path cost of 30 for VLANs 2 through 4, and configure the trunk port for Trunk 2 with a path cost of 30 for VLANs 8 through 10.

Configuring VMPS

The VLAN Query Protocol (VQP) supports dynamic-access ports, which are not permanently assigned to a VLAN, but give VLAN assignments based on the MAC source addresses seen on the port.

Note Because only UNIs can be configured as dynamic-access ports, only UNIs take part in VQP.

Each time an unknown MAC address is seen, the switch sends a VQP query to a remote VMPS; the query includes the newly seen MAC address and the port on which it was seen. The VMPS responds with a VLAN assignment for the port. The switch cannot be a VMPS server but can act as a client to the VMPS and communicate with it through VQP.

These sections contain this information:

- "Understanding VMPS" section on page 11-23
- "Default VMPS Client Configuration" section on page 11-24
- "VMPS Configuration Guidelines" section on page 11-25
- "Configuring the VMPS Client" section on page 11-25
- "Monitoring the VMPS" section on page 11-27
- "Troubleshooting Dynamic-Access Port VLAN Membership" section on page 11-28
- "VMPS Configuration Example" section on page 11-28

Understanding VMPS

Each time the client switch receives the MAC address of a new host, it sends a VQP query to the VMPS. When the VMPS receives this query, it searches its database for a MAC-address-to-VLAN mapping. The server response is based on this mapping and whether or not the server is in open or secure mode. In secure mode, the server shuts down the port when an illegal host is detected. In open mode, the server simply denies the host access to the port.

If the port is currently *unassigned* (that is, it does not yet have a VLAN assignment), the VMPS provides one of these responses:

- If the host is allowed on the port, the VMPS sends the client a *vlan-assignment* response containing the assigned VLAN name and allowing access to the host.
- If the host is not allowed on the port and the VMPS is in open mode, the VMPS sends an *access-denied* response.
- If the VLAN is not allowed on the port and the VMPS is in secure mode, the VMPS sends a *port-shutdown* response.

If the port already has a VLAN assignment, the VMPS provides one of these responses:

- If the VLAN in the database matches the current VLAN on the port, the VMPS sends an *success* response, allowing access to the host.
- If the VLAN in the database does not match the current VLAN on the port and active hosts exist on the port, the VMPS sends an *access-denied* or a *port-shutdown* response, depending on the secure mode of the VMPS.

If the switch receives an *access-denied* response from the VMPS, it continues to block traffic to and from the host MAC address. The switch continues to monitor the packets directed to the port and sends a query to the VMPS when it identifies a new host address. If the switch receives a *port-shutdown* response from the VMPS, it disables the port. The port must be manually re-enabled by using the CLI or SNMP.

Dynamic-Access Port VLAN Membership

A dynamic-access port can belong to only one VLAN with an ID from 1 to 4094. When the link comes up, the switch does not forward traffic to or from this port until the VMPS provides the VLAN assignment. The VMPS receives the source MAC address from the first packet of a new host connected to the dynamic-access port and attempts to match the MAC address to a VLAN in the VMPS database.

Note Only UNIs can be dynamic-access ports.

If there is a match, the VMPS sends the VLAN number for that port. If the client switch was not previously configured, it uses the domain name from the first VTP packet it receives on its trunk port from the VMPS. If the client switch was previously configured, it includes its domain name in the query packet to the VMPS to obtain its VLAN number. The VMPS verifies that the domain name in the packet matches its own domain name before accepting the request and responds to the client with the assigned VLAN number for the client. If there is no match, the VMPS either denies the request or shuts down the port (depending on the VMPS secure mode setting).

Multiple hosts (MAC addresses) can be active on a dynamic-access port if they are all in the same VLAN; however, the VMPS shuts down a dynamic-access port if more than 20 hosts are active on the port.

If the link goes down on a dynamic-access port, the port returns to an isolated state and does not belong to a VLAN. Any hosts that come online through the port are checked again through the VQP with the VMPS before the port is assigned to a VLAN.

Dynamic-access ports can be used for direct host connections, or they can connect to a network. A maximum of 20 MAC addresses are allowed per port on the switch. A dynamic-access port can belong to only one VLAN at a time, but the VLAN can change over time, depending on the MAC addresses seen.

Default VMPS Client Configuration

Table 11-6 shows the default VMPS and dynamic-access port configuration on client switches.

Table 11-6 Default VMPS Client and Dynamic-Access Port Configuration

Feature	Default Setting
VMPS domain server	None
VMPS reconfirm interval	60 minutes
VMPS server retry count	3
Dynamic-access ports	None configured

VMPS Configuration Guidelines

These guidelines and restrictions apply to dynamic-access port VLAN membership:

- You should configure the VMPS before you configure ports as dynamic-access ports.
- IEEE 802.1x ports cannot be configured as dynamic-access ports. If you try to enable IEEE 802.1x on a dynamic-access (VQP) port, an error message appears, and IEEE 802.1x is not enabled. If you try to change an IEEE 802.1x-enabled port to dynamic VLAN assignment, an error message appears, and the VLAN configuration is not changed.
- Trunk ports cannot be dynamic-access ports, but you can enter the **switchport access vlan dynamic** interface configuration command for a trunk port. In this case, the switch retains the setting and applies it if the port is later configured as an access port.

 You must turn off trunking on the port before the dynamic-access setting takes effect.
- Dynamic-access ports cannot be monitor ports.
- Secure ports cannot be dynamic-access ports. You must disable port security on a port before it becomes dynamic.
- Private VLAN ports cannot be dynamic-access ports.
- Dynamic-access ports cannot be members of an EtherChannel group.
- Port channels cannot be configured as dynamic-access ports.

Configuring the VMPS Client

You configure dynamic VLANs by using the VMPS (server). The switch can be a VMPS client; it cannot be a VMPS server.

Entering the IP Address of the VMPS

You must first enter the IP address of the server to configure the switch as a client.

Beginning in privileged EXEC mode, follow these steps to enter the IP address of the VMPS:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**vmps server** *ipaddress* **primary**	Enter the IP address of the switch acting as the primary VMPS server.
Step 3	**vmps server** *ipaddress*	(Optional) Enter the IP address of the switch acting as a secondary VMPS server. You can enter up to three secondary server addresses.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show vmps**	Verify your entries in the *VMPS Domain Server* field of the display.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Note You must have IP connectivity to the VMPS for dynamic-access ports to work. You can test for IP connectivity by pinging the IP address of the VMPS and verifying that you get a response.

Configuring Dynamic-Access Ports on VMPS Clients

Caution Dynamic-access port VLAN membership is for end stations or hubs connected to end stations. Connecting dynamic-access ports to other switches can cause a loss of connectivity.

Beginning in privileged EXEC mode, follow these steps to configure a dynamic-access port on a VMPS client switch:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the switch port that is connected to the end station, and enter interface configuration mode. The port must be a UNI.
Step 3n	**no shutdown**	Enable the port.
Step 4	**port-type uni**	Configure the port as a UNI.
Step 5	**switchport mode access**	Set the port to access mode.
Step 6	**switchport access vlan dynamic**	Configure the port as eligible for dynamic VLAN membership. The dynamic-access port must be connected to an end station.
Step 7	**end**	Return to privileged EXEC mode.
Step 8	**show interfaces** *interface-id* **switchport**	Verify your entries in the *Operational Mode* field of the display.
Step 9	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return an interface to its default configuration, use the **default interface** *interface-id* interface configuration command. To reset the access mode to the default VLAN for the switch, use the **no switchport access vlan** interface configuration command.

Reconfirming VLAN Memberships

Beginning in privileged EXEC mode, follow these steps to confirm the dynamic-access port VLAN membership assignments that the switch has received from the VMPS:

	Command	Purpose
Step 1	**vmps reconfirm**	Reconfirm dynamic-access port VLAN membership.
Step 2	**show vmps**	Verify the dynamic VLAN reconfirmation status.

Changing the Reconfirmation Interval

VMPS clients periodically reconfirm the VLAN membership information received from the VMPS.You can set the number of minutes after which reconfirmation occurs.

Beginning in privileged EXEC mode, follow these steps to change the reconfirmation interval:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**vmps reconfirm** *minutes*	Enter the number of minutes between reconfirmations of the dynamic VLAN membership. The range is 1 to 120. The default is 60 minutes.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show vmps**	Verify the dynamic VLAN reconfirmation status in the *Reconfirm Interval* field of the display.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the switch to its default setting, use the **no vmps reconfirm** global configuration command.

Changing the Retry Count

Beginning in privileged EXEC mode, follow these steps to change the number of times that the switch attempts to contact the VMPS before querying the next server:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**vmps retry** *count*	Change the retry count. The retry range is 1 to 10; the default is 3.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show vmps**	Verify your entry in the *Server Retry Count* field of the display.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the switch to its default setting, use the **no vmps retry** global configuration command.

Monitoring the VMPS

You can display information about the VMPS by using the **show vmps** privileged EXEC command. The switch displays this information about the VMPS:

- VMPS VQP Version—the version of VQP used to communicate with the VMPS. The switch queries the VMPS that is using VQP Version 1.
- Reconfirm Interval—the number of minutes the switch waits before reconfirming the VLAN-to-MAC-address assignments.
- Server Retry Count—the number of times VQP resends a query to the VMPS. If no response is received after this many tries, the switch starts to query the secondary VMPS.
- VMPS domain server—the IP address of the configured VLAN membership policy servers. The switch sends queries to the one marked *current*. The one marked *primary* is the primary server.

- VMPS Action—the result of the most recent reconfirmation attempt. A reconfirmation attempt can occur automatically when the reconfirmation interval expired, or you can force it by entering the **vmps reconfirm** privileged EXEC command.

This is an example of output for the **show vmps** privileged EXEC command:

```
Switch# show vmps
VQP Client Status:
--------------------
VMPS VQP Version:   1
Reconfirm Interval: 60 min
Server Retry Count: 3
VMPS domain server: 172.20.128.86 (primary, current)
                    172.20.128.87

Reconfirmation status
---------------------
VMPS Action:         other
```

Troubleshooting Dynamic-Access Port VLAN Membership

The VMPS shuts down a dynamic-access port under these conditions:

- The VMPS is in secure mode, and it does not allow the host to connect to the port. The VMPS shuts down the port to prevent the host from connecting to the network.
- More than 20 active hosts reside on a dynamic-access port.

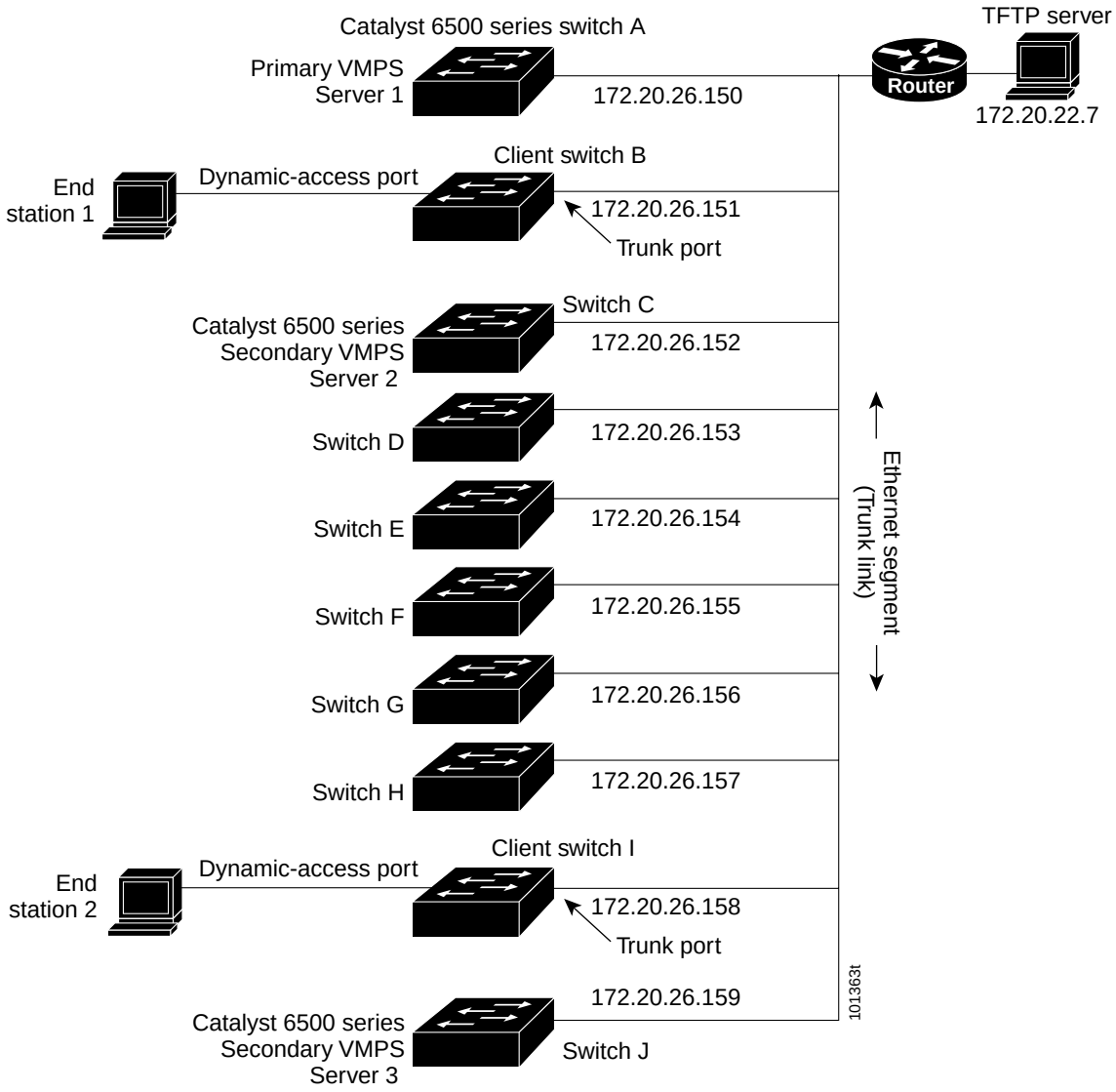
To disable and re-enable a disabled dynamic-access port, enter the **shutdown** interface configuration command followed by the **no shutdown** interface configuration command.

VMPS Configuration Example

Figure 11-5 shows a network with a VMPS server switch and VMPS client switches with dynamic-access ports. In this example, these assumptions apply:

- The VMPS server and the VMPS client are separate switches.
- The Catalyst 6500 series Switch A is the primary VMPS server.
- The Catalyst 6500 series Switch C and Switch J are secondary VMPS servers.
- End stations are connected to the clients, Switch B and Switch I.
- The database configuration file is stored on the TFTP server with the IP address 172.20.22.7.

Figure 11-5 Dynamic Port VLAN Membership Configuration





CHAPTER 12

Configuring Private VLANs

This chapter describes how to configure private VLANs on the Cisco ME 3400 Ethernet Access switch.

Note For complete syntax and usage information for the commands used in this chapter, see the command reference for this release.

The chapter consists of these sections:

- Understanding Private VLANs, page 12-1
- Configuring Private VLANs, page 12-5
- Monitoring Private VLANs, page 12-15

Understanding Private VLANs

The private-VLAN feature addresses two problems that service providers face when using VLANs:

- Scalability: The switch supports up to 1005 active VLANs. If a service provider assigns one VLAN per customer, this limits the numbers of customers that the service provider can support.
- To enable IP routing, each VLAN is assigned a subnet address space or a block of addresses, which can result in wasting the unused IP addresses and cause IP address management problems.

Using private VLANs addresses the scalability problem and provides IP address management benefits for service providers and Layer 2 security for customers.

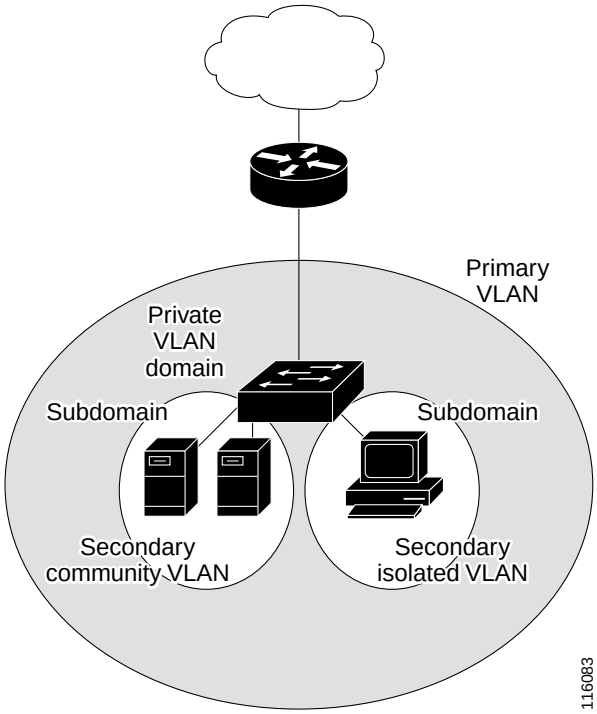
These sections describe how private VLANs work:

- Types of Private VLANs and Private-VLAN Ports, page 12-2
- IP Addressing Scheme with Private VLANs, page 12-4
- Private VLANs across Multiple Switches, page 12-4
- Private VLANs and Unicast, Broadcast, and Multicast Traffic, page 12-5
- Private VLANs and SVIs, page 12-5

Types of Private VLANs and Private-VLAN Ports

Private VLANs partition a regular VLAN domain into subdomains. A subdomain is represented by a pair of VLANs: a *primary* VLAN and a *secondary* VLAN. A private VLAN can have multiple VLAN pairs, one pair for each subdomain. All VLAN pairs in a private VLAN share the same primary VLAN. The secondary VLAN ID differentiates one subdomain from another. See Figure 12-1.

Figure 12-1 Private-VLAN Domain



There are two types of secondary VLANs:

- Isolated VLANs—Ports within an isolated VLAN cannot communicate with each other at the Layer 2 level.
- Community VLANs—Ports within a community VLAN can communicate with each other but cannot communicate with ports in other communities at the Layer 2 level. A community VLAN can include no more than eight user network interfaces (UNIs).

Private VLANs provide Layer 2 isolation between ports within the same private VLAN. Private-VLAN ports are access ports that are one of these types:

- Promiscuous—A promiscuous port belongs to the primary VLAN and can communicate with all interfaces, including the community and isolated host ports that belong to the secondary VLANs associated with the primary VLAN.

Note Promiscuous ports must be network node interfaces (NNIs). UNIs cannot be configured as promiscuous ports.

- Isolated—An isolated port is a host port that belongs to an isolated secondary VLAN. It has complete Layer 2 separation from other ports within the same private VLAN, except for the promiscuous ports. Private VLANs block all traffic to isolated ports except traffic from promiscuous ports. Traffic received from an isolated port is forwarded only to promiscuous ports.
- Community—A community port is a host port that belongs to a community secondary VLAN. Community ports communicate with other ports in the same community VLAN and with promiscuous ports. These interfaces are isolated at Layer 2 from all other interfaces in other communities and from isolated ports within their private VLAN. No more than eight UNIs can be community ports in the same community VLAN.

Note Trunk ports carry traffic from regular VLANs and also from primary, isolated, and community VLANs.

Primary and secondary VLANs have these characteristics:

- Primary VLAN—A private VLAN has only one primary VLAN. Every port in a private VLAN is a member of the primary VLAN. The primary VLAN carries unidirectional traffic downstream from the promiscuous ports to the (isolated and community) host ports and to other promiscuous ports.
- Isolated VLAN —A private VLAN has only one isolated VLAN. An isolated VLAN is a secondary VLAN that carries unidirectional traffic upstream from the hosts toward the promiscuous ports and the gateway.
- Community VLAN—A community VLAN is a secondary VLAN that carries upstream traffic from the community ports to the promiscuous port gateways and to other host ports in the same community. You can configure multiple community VLANs in a private VLAN. Each community VLAN can include no more than eight UNIs.

Note The switch also supports UNI isolated VLANs and UNI community VLANs. When a VLAN is created, it is by default a UNI isolated VLAN. Traffic is not switched among UNIs on a switch that belong to a UNI isolated VLAN. For more information on UNI VLANs, see Chapter 11, “Configuring VLANs.”

A promiscuous port can serve only one primary VLAN, one isolated VLAN, and multiple community VLANs. Layer 3 gateways are typically connected to the switch through a promiscuous port. With a promiscuous port, you can connect a wide range of devices as access points to a private VLAN. For example, you can use a promiscuous port to monitor or back up all the private-VLAN servers from an administration workstation.

In a switched environment, you can assign an individual private VLAN and associated IP subnet to each individual or common group of end stations. The end stations need to communicate only with a default gateway to communicate outside the private VLAN.

You can use private VLANs to control access to end stations in these ways:

- Configure selected interfaces connected to end stations as isolated ports to prevent any communication at Layer 2. For example, if the end stations are servers, this configuration prevents Layer 2 communication between the servers.
- Configure NNIs connected to default gateways and selected end stations (for example, backup servers) as promiscuous ports to allow all end stations access to a default gateway.

You can extend private VLANs across multiple devices by trunking the primary, isolated, and community VLANs to other devices that support private VLANs. To maintain the security of your private-VLAN configuration and to avoid other use of the VLANs configured as private VLANs, configure private VLANs on all intermediate devices, including devices that have no private-VLAN ports.

IP Addressing Scheme with Private VLANs

Assigning a separate VLAN to each customer creates an inefficient IP addressing scheme:

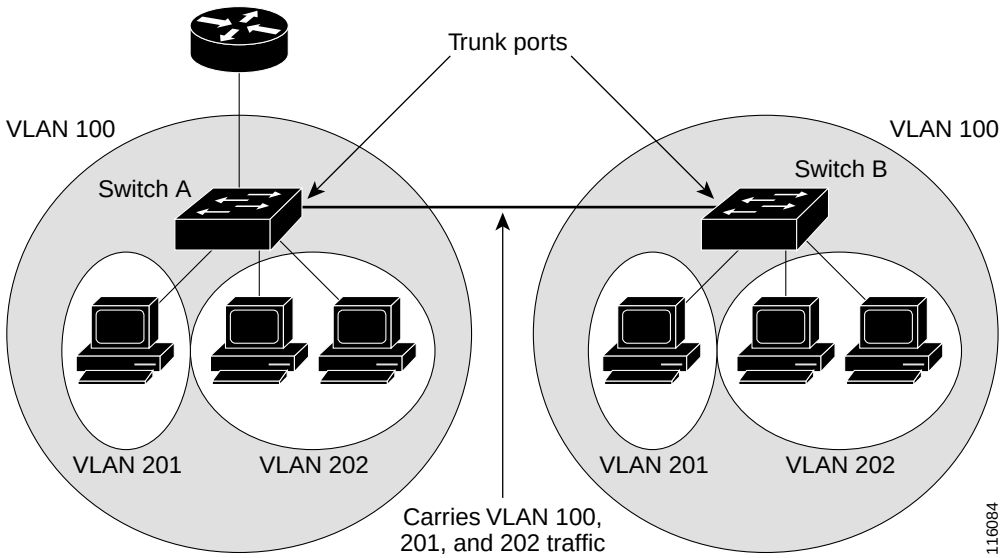
- Assigning a block of addresses to a customer VLAN can result in unused IP addresses.
- If the number of devices in the VLAN increases, the number of assigned address might not be large enough to accommodate them.

These problems are reduced by using private VLANs, where all members in the private VLAN share a common address space, which is allocated to the primary VLAN. Hosts are connected to secondary VLANs, and the DHCP server assigns them IP addresses from the block of addresses allocated to the primary VLAN. Subsequent IP addresses can be assigned to customer devices in different secondary VLANs, but in the same primary VLAN. When new devices are added, the DHCP server assigns them the next available address from a large pool of subnet addresses.

Private VLANs across Multiple Switches

As with regular VLANs, private VLANs can span multiple switches. A trunk port carries the primary VLAN and secondary VLANs to a neighboring switch. The trunk port treats the private VLAN as any other VLAN. A feature of private VLANs across multiple switches is that traffic from an isolated port in switch A does not reach an isolated port on Switch B. See Figure 12-2.

Figure 12-2 Private VLANs across Switches



VLAN 100 = Primary VLAN
VLAN 201 = Secondary isolated VLAN
VLAN 202 = Secondary community VLAN

You must manually configure private VLANs on all switches in the Layer 2 network. If you do not configure the primary and secondary VLAN association in some switches in the network, the Layer 2 databases in these switches are not merged. This can result in unnecessary flooding of private-VLAN traffic on those switches.

Private VLANs and Unicast, Broadcast, and Multicast Traffic

In regular VLANs, devices in the same VLAN can communicate with each other at the Layer 2 level, but devices connected to interfaces in different VLANs must communicate at the Layer 3 level. In private VLANs, the promiscuous ports are members of the primary VLAN, while the host ports belong to secondary VLANs. Because the secondary VLAN is associated to the primary VLAN, members of the these VLANs can communicate with each other at the Layer 2 level.

In a regular VLAN, broadcasts are forwarded to all ports in that VLAN. Private-VLAN broadcast forwarding depends on the port sending the broadcast:

- An isolated port sends a broadcast only to the promiscuous ports or trunk ports.
- A community port sends a broadcast to all promiscuous ports, trunk ports, and ports in the same community VLAN.
- A promiscuous port (only NNI) sends a broadcast to all ports in the private VLAN (other promiscuous ports, trunk ports, isolated ports, and community ports).

Multicast traffic is routed or bridged across private-VLAN boundaries and within a single community VLAN. Multicast traffic is not forwarded between ports in the same isolated VLAN or between ports in different secondary VLANs.

Private VLANs and SVIs

In a Layer 3 switch (a switch running the metro IP access image), a switch virtual interface (SVI) represents the Layer 3 interface of a VLAN. Layer 3 devices communicate with a private VLAN only through the primary VLAN and not through secondary VLANs. Configure Layer 3 VLAN interfaces (SVIs) only for primary VLANs. You cannot configure Layer 3 VLAN interfaces for secondary VLANs. SVIs for secondary VLANs are inactive while the VLAN is configured as a secondary VLAN.

- If you try to configure a VLAN with an active SVI as a secondary VLAN, the configuration is not allowed until you disable the SVI.
- If you try to create an SVI on a VLAN that is configured as a secondary VLAN and the secondary VLAN is already mapped at Layer 3, the SVI is not created, and an error is returned. If the SVI is not mapped at Layer 3, the SVI is created, but it is automatically shut down.

When the primary VLAN is associated with and mapped to the secondary VLAN, any configuration on the primary VLAN is propagated to the secondary VLAN SVIs. For example, if you assign an IP subnet to the primary VLAN SVI, this subnet is the IP subnet address of the entire private VLAN.

Configuring Private VLANs

These sections contain this configuration information:

- Tasks for Configuring Private VLANs, page 12-6
- Default Private-VLAN Configuration, page 12-6
- Private-VLAN Configuration Guidelines, page 12-6
- Configuring and Associating VLANs in a Private VLAN, page 12-10
- Configuring a Layer 2 Interface as a Private-VLAN Host Port, page 12-12

- Configuring a Layer 2 Interface as a Private-VLAN Promiscuous Port, page 12-13
- Mapping Secondary VLANs to a Primary VLAN Layer 3 VLAN Interface, page 12-14

Tasks for Configuring Private VLANs

To configure a private VLAN, perform these steps:

Step 1 Create the primary and secondary VLANs and associate them. See the "Configuring and Associating VLANs in a Private VLAN" section on page 12-10.

Note If the VLAN is not created already, the private-VLAN configuration process creates it.

Step 2 Configure interfaces to be isolated or community host ports, and assign VLAN membership to the host port. See the "Configuring a Layer 2 Interface as a Private-VLAN Host Port" section on page 12-12.

Step 3 Configure NNIs as promiscuous ports, and map the promiscuous ports to the primary-secondary VLAN pair. See the "Configuring a Layer 2 Interface as a Private-VLAN Promiscuous Port" section on page 12-13.

Step 4 If inter-VLAN routing will be used, configure the primary SVI, and map secondary VLANs to the primary. See the "Mapping Secondary VLANs to a Primary VLAN Layer 3 VLAN Interface" section on page 12-14.

Step 5 Verify private-VLAN configuration.

Default Private-VLAN Configuration

No private VLANs are configured. Newly created VLANs are UNI isolated VLANs.

Private-VLAN Configuration Guidelines

Guidelines for configuring private VLANs fall into these categories:

- Secondary and Primary VLAN Configuration, page 12-7
- Private-VLAN Port Configuration, page 12-8
- Limitations with Other Features, page 12-9

Secondary and Primary VLAN Configuration

Follow these guidelines when configuring private VLANs:

- You use VLAN configuration mode to configure private VLANs. For more information about VLAN configuration, see the "Creating and Modifying VLANs" section on page 11-6.
- You must configure private VLANs on each device where you want private-VLAN ports.
- A private VLAN cannot be a UNI VLAN.
 - To change a UNI isolated VLAN (the default) to a private VLAN, enter the **private-vlan** VLAN configuration command; this overwrites the default isolated VLAN configuration.
 - To change a UNI community VLAN to a private VLAN, you must first enter the **no uni-vlan** VLAN configuration command to return to the default UNI isolated VLAN configuration.
- You cannot configure VLAN 1 or VLANs 1002 to 1005 as primary or secondary VLANs. Extended VLANs (VLAN IDs 1006 to 4094) can belong to private VLANs
- A primary VLAN can have one isolated VLAN and multiple community VLANs associated with it. An isolated or community VLAN can have only one primary VLAN associated with it.
- Although a private VLAN contains more than one VLAN, only one Spanning Tree Protocol (STP) instance runs for the entire private VLAN. When a secondary VLAN is associated with the primary VLAN, the STP parameters of the primary VLAN are propagated to the secondary VLAN.
- You can enable DHCP snooping on private VLANs. When you enable DHCP snooping on the primary VLAN, it is propagated to the secondary VLANs. If you configure DHCP on a secondary VLAN, the configuration does not take effect if the primary VLAN is already configured.
- If the switch is running the metro access or metro IP access image and you enable IP source guard on private-VLAN ports, you must enable DHCP snooping on the primary VLAN.
- You can apply different quality of service (QoS) configurations to primary, isolated, and community VLANs.
- When the switch is running the metro IP access image and you configure private VLANs, sticky Address Resolution Protocol (ARP) is enabled by default, and ARP entries learned on Layer 3 private VLAN interfaces are sticky ARP entries. For security reasons, private VLAN port sticky ARP entries do not age out.

 Note We recommend that you display and verify private-VLAN interface ARP entries.

 Connecting a device with a different MAC address but with the same IP address displays a message, and the ARP entry is not created. Because the private-VLAN port sticky ARP entries do not age out, you must manually remove private-VLAN port ARP entries if a MAC address changes.
 - You can remove a private-VLAN ARP entry by using the **no arp** *ip-address* global configuration command.
 - You can add a private-VLAN ARP entry by using the **arp** ip-*address hardware-address* **type** global configuration command.
- You can configure VLAN maps on primary and secondary VLANs (see the "Configuring VLAN Maps" section on page 28-29). However, we recommend that you configure the same VLAN maps on private-VLAN primary and secondary VLANs.

- When a frame is Layer 2 forwarded within a private VLAN, the same VLAN map is applied at the receiving and sending sides. When a frame is routed from inside a private VLAN to an external port, the private-VLAN map is applied at the receiving side.
 - For frames going upstream from a host port to a promiscuous port, the VLAN map configured on the secondary VLAN is applied.
 - For frames going downstream from a promiscuous port to a host port, the VLAN map configured on the primary VLAN is applied.

 To filter out specific IP traffic for a private VLAN, you should apply the VLAN map to both the primary and secondary VLANs.
- If the switch is running the metro IP access image, you can apply router ACLs only on the primary-VLAN SVIs. The ACL is applied to both primary and secondary VLAN Layer 3 traffic.
- Although private VLANs provide host isolation at Layer 2, hosts can communicate with each other at Layer 3.
- Private VLANs support these Switched Port Analyzer (SPAN) features:
 - You can configure a private-VLAN port as a SPAN source port.
 - You can use VLAN-based SPAN (VSPAN) on primary, isolated, and community VLANs or use SPAN on only one VLAN to separately monitor sent or received traffic.

Private-VLAN Port Configuration

Follow these guidelines when configuring private-VLAN ports:

- Promiscuous ports must be NNIs; UNIs cannot be configured as promiscuous ports.
- Use only the private-VLAN configuration commands to assign ports to primary, isolated, or community VLANs. Layer 2 access ports assigned to the VLANs that you configure as primary, isolated, or community VLANs are inactive while the VLAN is part of the private-VLAN configuration. Layer 2 trunk interfaces remain in the STP forwarding state.
- Do not configure NNI ports that belong to a Port Aggregation Protocol (PAgP) or Link Aggregation Control Protocol (LACP) EtherChannel as private-VLAN ports. While a port is part of the private-VLAN configuration, any EtherChannel configuration for it is inactive.
- Enable Port Fast and BPDU guard on NNI isolated and community host ports to prevent STP loops due to misconfigurations and to speed up STP convergence (see Chapter 16, “Configuring Optional Spanning-Tree Features”). When enabled, STP applies the BPDU guard feature to all Port Fast-configured Layer 2 LAN ports. Do not enable Port Fast and BPDU guard on promiscuous ports.
- If you delete a VLAN used in the private-VLAN configuration, the private-VLAN ports associated with the VLAN become inactive.
- Private-VLAN ports can be on different network devices if the devices are trunk-connected and the primary and secondary VLANs have not been removed from the trunk.
- A community private VLAN can include no more than eight UNIs. If you try to add more than eight UNIs, the configuration is not allowed. If you try to configure a VLAN that includes more than eight UNIs as a community private VLAN, the configuration is not allowed.

Limitations with Other Features

When configuring private VLANs, remember these limitations with other features:

Note In some cases, the configuration is accepted with no error messages, but the commands have no effect.

- When IGMP snooping is enabled on the switch (the default), the switch supports no more than 20 private-VLAN domains.
- A private VLAN cannot be a UNI isolated or UNI community VLAN. For more information about UNI VLANs, see Chapter 11, "Configuring VLANs."
- Do not configure a remote SPAN (RSPAN) VLAN as a private-VLAN primary or secondary VLAN. For more information about SPAN, see Chapter 24, "Configuring SPAN and RSPAN."
- Do not configure private-VLAN ports on interfaces configured for these other features:
 - dynamic-access port VLAN membership
 - PAgP (only NNIs)
 - LACP (only NNIs)
 - Multicast VLAN Registration (MVR)
- A private-VLAN port cannot be a secure port, and an NNI should not be configured as a protected port.
- You can configure 802.1x port-based authentication on a private-VLAN port, but do not configure IEEE 802.1x with port security on private-VLAN ports.
- A private-VLAN host or promiscuous port cannot be a SPAN destination port. If you configure a SPAN destination port as a private-VLAN port, the port becomes inactive.
- If you configure a static MAC address on a promiscuous port in the primary VLAN, you must add the same static address to all associated secondary VLANs. If you configure a static MAC address on a host port in a secondary VLAN, you must add the same static MAC address to the associated primary VLAN. When you delete a static MAC address from a private-VLAN port, you must remove all instances of the configured MAC address from the private VLAN.

 Note Dynamic MAC addresses learned in one VLAN of a private VLAN are replicated in the associated VLANs. For example, a MAC address learned in a secondary VLAN is replicated in the primary VLAN. When the original dynamic MAC address is deleted or aged out, the replicated addresses are removed from the MAC address table.

- Configure Layer 3 VLAN interfaces (SVIs) only for primary VLANs.

Configuring and Associating VLANs in a Private VLAN

Beginning in privileged EXEC mode, follow these steps to configure a private VLAN:

Note The **private-vlan** commands do not take effect until you exit VLAN configuration mode.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**vlan** *vlan-id*	Enter VLAN configuration mode, and designate or create a VLAN that will be the primary VLAN. The VLAN ID range is 2 to 1001 and 1006 to 4094. **Note** If the VLAN has been configured as a UNI community VLAN, you must enter the **no uni-vlan** VLAN configuration command before configuring a private VLAN.
Step 3	**private-vlan primary**	Designate the VLAN as the primary VLAN.
Step 4	**exit**	Return to global configuration mode.
Step 5	**vlan** *vlan-id*	(Optional) Enter VLAN configuration mode, and designate or create a VLAN that will be an isolated VLAN. The VLAN ID range is 2 to 1001 and 1006 to 4094.
Step 6	**private-vlan isolated**	Designate the VLAN as an isolated VLAN.
Step 7	**exit**	Return to global configuration mode.
Step 8	**vlan** *vlan-id*	(Optional) Enter VLAN configuration mode. and designate or create a VLAN that will be a community VLAN. The VLAN ID range is 2 to 1001 and 1006 to 4094. **Note** If the VLAN has been configured as a UNI community VLAN, you must enter the **no uni-vlan** VLAN configuration command before configuring a private VLAN.
Step 9	**private-vlan community**	Designate the VLAN as a community VLAN.
Step 10	**exit**	Return to global configuration mode.
Step 11	**vlan** *vlan-id*	Enter VLAN configuration mode for the primary VLAN designated in Step 3.
Step 12	**private-vlan association** [**add** \| **remove**] *secondary_vlan_list*	Associate the secondary VLANs with the primary VLAN.
Step 13	**end**	Return to privileged EXEC mode.
Step 14	**show vlan private-vlan** [**type**] or **show interfaces status**	Verify the configuration.
Step 15	**copy running-config startup config**	(Optional) Save your entries in the switch startup configuration file.

When you associate secondary VLANs with a primary VLAN, note this syntax information:

- The *secondary_vlan_list* parameter cannot contain spaces. It can contain multiple comma-separated items. Each item can be a single private-VLAN ID or a hyphenated range of private-VLAN IDs.
- The *secondary_vlan_list* parameter can contain multiple community VLAN IDs but only one isolated VLAN ID.
- Enter a *secondary_vlan_list*, or use the **add** keyword with a *secondary_vlan_list* to associate secondary VLANs with a primary VLAN.
- Use the **remove** keyword with a *secondary_vlan_list* to clear the association between secondary VLANs and a primary VLAN.
- The **private-vlan association** VLAN configuration command does not take effect until you exit VLAN configuration mode.

This example shows how to configure VLAN 20 as a primary VLAN, VLAN 501 as an isolated VLAN, and VLANs 502 and 503 as community VLANs, to associate them in a private VLAN, and to verify the configuration. It assumes that VLANs 502 and 503 have previously been configured as UNI community VLANs:

```
Switch# configure terminal
Switch(config)# vlan 20
Switch(config-vlan)# private-vlan primary
Switch(config-vlan)# exit
Switch(config)# vlan 501
Switch(config-vlan)# private-vlan isolated
Switch(config-vlan)# exit
Switch(config)# vlan 502
Switch(config-vlan)# no-uni vlan
Switch(config-vlan)# private-vlan community
Switch(config-vlan)# exit
Switch(config)# vlan 503
Switch(config-vlan)# no-uni vlan
Switch(config-vlan)# private-vlan community
Switch(config-vlan)# exit
Switch(config)# vlan 20
Switch(config-vlan)# private-vlan association 501-503
Switch(config-vlan)# end
Switch(config)# show vlan private vlan
Primary Secondary Type              Ports
------- --------- ----------------- ------------------------------------------
20      501       isolated
20      502       community
20      503       community
20      504       non-operational
```

Configuring a Layer 2 Interface as a Private-VLAN Host Port

Beginning in privileged EXEC mode, follow these steps to configure a Layer 2 interface as a private-VLAN host port and to associate it with primary and secondary VLANs:

Note Isolated and community VLANs are both secondary VLANs.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode for the Layer 2 interface to be configured.
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled, and NNIs are enabled.
Step 4	**switchport mode private-vlan host**	Configure the Layer 2 port as a private-VLAN host port.
Step 5	**switchport private-vlan host-association** *primary_vlan_id secondary_vlan_id*	Associate the Layer 2 port with a private VLAN.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show interfaces** [*interface-id*] **switchport**	Verify the configuration.
Step 8	**copy running-config startup config**	(Optional) Save your entries in the switch startup configuration file.

This example shows how to configure an interface as a private-VLAN host port, associate it with a private-VLAN pair, and verify the configuration:

```
Switch# configure terminal
Switch(config)# interface fastethernet0/22
Switch(config-if)# no shutdown
Switch(config-if)# switchport mode private-vlan host
Switch(config-if)# switchport private-vlan host-association 20 25
Switch(config-if)# end
Switch# show interfaces fastethernet0/22 switchport
Name: Fa0/22
Switchport: Enabled
Administrative Mode: private-vlan host
Operational Mode: private-vlan host
Administrative Trunking Encapsulation: dot1q
Operational Trunking Encapsulation: native
Negotiation of Trunking: Off
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 1 (default)
Administrative Native VLAN tagging: enabled
Administrative private-vlan host-association: 20 (VLAN0020) 25 (VLAN0025)
Administrative private-vlan mapping: none
Administrative private-vlan trunk native VLAN: none
Administrative private-vlan trunk Native VLAN tagging: enabled
Administrative private-vlan trunk encapsulation: dot1q
Administrative private-vlan trunk normal VLANs: none
Administrative private-vlan trunk private VLANs: none
Operational private-vlan:
20 (VLAN0020) 25 (VLAN0025)
<output truncated>
```

Configuring a Layer 2 Interface as a Private-VLAN Promiscuous Port

You can configure only NNIs as promiscuous ports. Beginning in privileged EXEC mode, follow these steps to configure a Layer 2 interface as a private-VLAN promiscuous port and map it to primary and secondary VLANs:

Note Isolated and community VLANs are both secondary VLANs.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode for the Layer 2 interface to be configured. The interface must be an NNI. **Note** If the interface is a UNI, you must enter the **port-type nni** interface configuration command before configuring it as a promiscuous port.
Step 3	**switchport mode private-vlan promiscuous**	Configure the Layer 2 NNI port as a private-VLAN promiscuous port.
Step 4	**switchport private-vlan mapping** *primary_vlan_id* {**add** \| **remove**} *secondary_vlan_list*	Map the private-VLAN promiscuous port to a primary VLAN and to selected secondary VLANs.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show interfaces** [*interface-id*] **switchport**	Verify the configuration.
Step 7	**copy running-config startup config**	(Optional) Save your entries in the switch startup configuration file.

When you configure a Layer 2 interface as a private-VLAN promiscuous port, note this syntax information:

- The *secondary_vlan_list* parameter cannot contain spaces. It can contain multiple comma-separated items. Each item can be a single private-VLAN ID or a hyphenated range of private-VLAN IDs.
- Enter a *secondary_vlan_list*, or use the **add** keyword with a *secondary_vlan_list* to map the secondary VLANs to the private-VLAN promiscuous port.
- Use the **remove** keyword with a *secondary_vlan_list* to clear the mapping between secondary VLANs and the private-VLAN promiscuous port.

This example shows how to configure an NNI as a private-VLAN promiscuous port and map it to a private VLAN. The interface is a member of primary VLAN 20 and secondary VLANs 501 to 503 are mapped to it.

```
Switch# configure terminal
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# switchport mode private-vlan promiscuous
Switch(config-if)# switchport private-vlan mapping 20 add 501-503
Switch(config-if)# end
```

Use the **show vlan private-vlan** or the **show interface status** privileged EXEC command to display primary and secondary VLANs and private-VLAN ports on the switch.

Mapping Secondary VLANs to a Primary VLAN Layer 3 VLAN Interface

If the switch is running the metro IP access image and the private VLAN will be used for inter-VLAN routing, you configure an SVI for the primary VLAN and map secondary VLANs to the SVI.

Note Isolated and community VLANs are both secondary VLANs.

Beginning in privileged EXEC mode, follow these steps to map secondary VLANs to the SVI of a primary VLAN to allow Layer 3 switching of private-VLAN traffic:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface vlan** *primary_vlan_id*	Enter interface configuration mode for the primary VLAN, and configure the VLAN as an SVI. The VLAN ID range is 2 to 1001 and 1006 to 4094.
Step 3	**private-vlan mapping** [**add** \| **remove**] *secondary_vlan_list*	Map the secondary VLANs to the Layer 3 VLAN interface of a primary VLAN to allow Layer 3 switching of private-VLAN incoming traffic.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show interface private-vlan mapping**	Verify the configuration.
Step 6	**copy running-config startup config**	(Optional) Save your entries in the switch startup configuration file.

Note The **private-vlan mapping** interface configuration command only affects private-VLAN traffic that is Layer 3 switched.

When you map secondary VLANs to the Layer 3 VLAN interface of a primary VLAN, note this syntax information:

- The *secondary_vlan_list* parameter cannot contain spaces. It can contain multiple comma-separated items. Each item can be a single private-VLAN ID or a hyphenated range of private-VLAN IDs.
- Enter a *secondary_vlan_list,* or use the **add** keyword with a *secondary_vlan_list* to map the secondary VLANs to the primary VLAN.
- Use the **remove** keyword with a *secondary_vlan_list* to clear the mapping between secondary VLANs and the primary VLAN.

This example shows how to map the interfaces of VLANs 501and 502 to primary VLAN 10, which permits routing of secondary VLAN incoming traffic from private VLANs 501 to 502:

```
Switch# configure terminal
Switch(config)# interface vlan 10
Switch(config-if)# private-vlan mapping 501-502
Switch(config-if)# end
Switch# show interfaces private-vlan mapping
Interface Secondary VLAN Type
--------- -------------- -----------------
vlan10    501            isolated
vlan10    502            community
```

Monitoring Private VLANs

Table 12-1 shows the privileged EXEC commands for monitoring private-VLAN activity.

Table 12-1 Private VLAN Monitoring Commands

Command	Purpose
show interfaces status	Displays the status of interfaces, including the VLANs to which they belong.
show vlan private-vlan [**type**]	Display the private-VLAN information for the switch.
show interface switchport	Display the private-VLAN configuration on interfaces.
show interface private-vlan mapping	Display information about the private-VLAN mapping for VLAN interfaces.

This is an example of the output from the **show vlan private-vlan** command:

```
Switch(config)# show vlan private-vlan
Primary Secondary Type              Ports
------- --------- ----------------- ------------------------------------------
10      501       isolated          Fa0/1, Gi0/1, Gi0/2
10      502       community         Fa0/11, Fa0/12, Gi0/1
10      503       non-operational
```



CHAPTER 13

Configuring IEEE 802.1Q and Layer 2 Protocol Tunneling

Virtual private networks (VPNs) provide enterprise-scale connectivity on a shared infrastructure, often Ethernet-based, with the same security, prioritization, reliability, and manageability requirements of private networks. Tunneling is a feature designed for service providers who carry traffic of multiple customers across their networks and are required to maintain the VLAN and Layer 2 protocol configurations of each customer without impacting the traffic of other customers. The Cisco ME 3400 Ethernet Access switch supports IEEE 802.1Q tunneling and Layer 2 protocol tunneling when it is running the metro access or metro IP access image. The metro base image does not support tunneling.

Note For complete syntax and usage information for the commands used in this chapter, see the command reference for this release.

This chapter contains these sections:

- Understanding IEEE 802.1Q Tunneling, page 13-1
- Configuring IEEE 802.1Q Tunneling, page 13-4
- Understanding Layer 2 Protocol Tunneling, page 13-8
- Configuring Layer 2 Protocol Tunneling, page 13-10
- Monitoring and Maintaining Tunneling Status, page 13-18

Understanding IEEE 802.1Q Tunneling

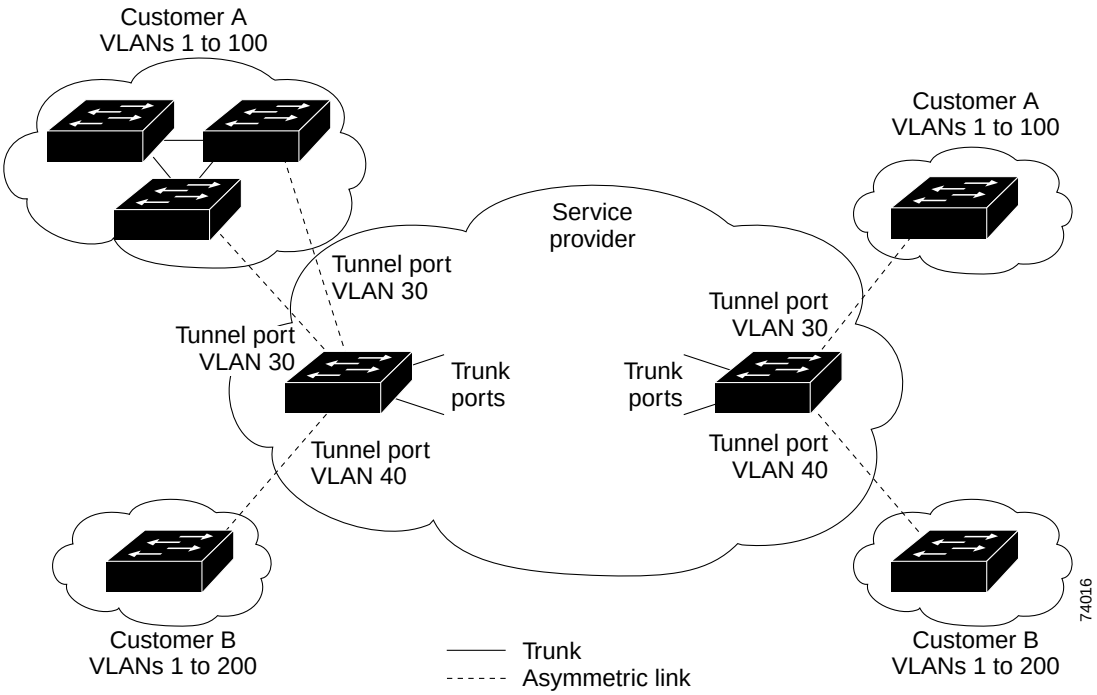
Business customers of service providers often have specific requirements for VLAN IDs and the number of VLANs to be supported. The VLAN ranges required by different customers in the same service-provider network might overlap, and traffic of customers through the infrastructure might be mixed. Assigning a unique range of VLAN IDs to each customer would restrict customer configurations and could easily exceed the VLAN limit (4096) of the IEEE 802.1Q specification.

Using the IEEE 802.1Q tunneling feature, service providers can use a single VLAN to support customers who have multiple VLANs. Customer VLAN IDs are preserved, and traffic from different customers is segregated within the service-provider network, even when they appear to be in the same VLAN. Using IEEE 802.1Q tunneling expands VLAN space by using a VLAN-in-VLAN hierarchy and retagging the tagged packets. A port configured to support IEEE 802.1Q tunneling is called a *tunnel port*. When you configure tunneling, you assign a tunnel port to a VLAN ID that is dedicated to tunneling. Each customer requires a separate service-provider VLAN ID, but that VLAN ID supports all of the customer's VLANs.

Customer traffic tagged in the normal way with appropriate VLAN IDs comes from an IEEE 802.1Q trunk port on the customer device and into a tunnel port on the service-provider edge switch. The link between the customer device and the edge switch is asymmetric because one end is configured as an IEEE 802.1Q trunk port, and the other end is configured as a tunnel port. You assign the tunnel port interface to an access VLAN ID that is unique to each customer. See Figure 13-1.

Note By default, VLANs configured on the switch are user network interface (UNI) isolated VLANs. In a UNI isolated VLAN, IEEE 802.1Q tunneled access ports on the switch are isolated from each other. If you use the **uni-vlan community** VLAN configuration command to change a VLAN to a UNI community VLAN, local switching occurs between these ports. For more information about UNI VLANs, see Chapter 11, "Configuring VLANs."

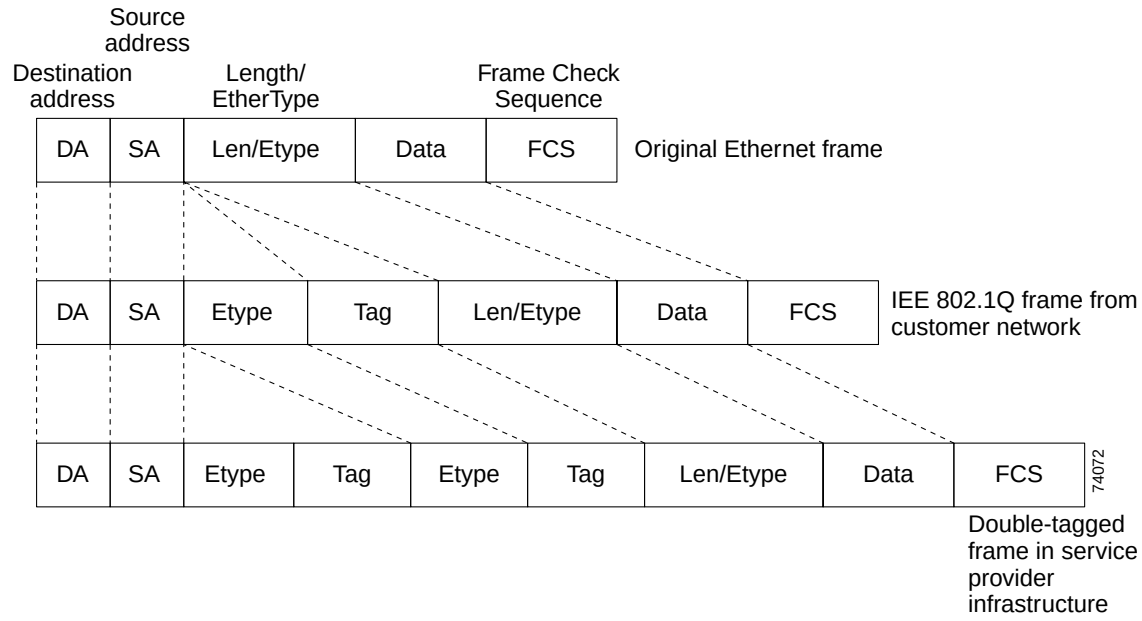
Figure 13-1 IEEE 802.1Q Tunnel Ports in a Service-Provider Network



Packets coming from the customer trunk port into the tunnel port on the service-provider edge switch are normally IEEE 802.1Q-tagged with the appropriate VLAN ID. The the tagged packets remain intact inside the switch and when they exit the trunk port into the service-provider network, they are encapsulated with another layer of an IEEE 802.1Q tag (called the *metro tag*) that contains the VLAN ID that is unique to the customer. The original customer IEEE 802.1Q tag is preserved in the encapsulated packet. Therefore, packets entering the service-provider network are double-tagged, with the outer (metro) tag containing the customer's access VLAN ID, and the inner VLAN ID being that of the incoming traffic.

When the double-tagged packet enters another trunk port in a service-provider core switch, the outer tag is stripped as the switch processes the packet. When the packet exits another trunk port on the same core switch, the same metro tag is again added to the packet. Figure 13-2 shows the tag structures of the double-tagged packets.

Figure 13-2 Original (Normal), IEEE 802.1Q, and Double-Tagged Ethernet Packet Formats



When the packet enters the trunk port of the service-provider egress switch, the outer tag is again stripped as the switch internally processes the packet. However, the metro tag is not added when the packet is sent out the tunnel port on the edge switch into the customer network. The packet is sent as a normal IEEE 802.1Q-tagged frame to preserve the original VLAN numbers in the customer network.

In Figure 13-1, Customer A was assigned VLAN 30, and Customer B was assigned VLAN 40. Packets entering the edge switch tunnel ports with IEEE 802.1Q tags are double-tagged when they enter the service-provider network, with the outer tag containing VLAN ID 30 or 40, appropriately, and the inner tag containing the original VLAN number, for example, VLAN 100. Even if both Customers A and B have VLAN 100 in their networks, the traffic remains segregated within the service-provider network because the outer tag is different. Each customer controls its own VLAN numbering space, which is independent of the VLAN numbering space used by other customers and the VLAN numbering space used by the service-provider network.

At the outbound tunnel port, the original VLAN numbers on the customer's network are recovered. It is possible to have multiple levels of tunneling and tagging, but the switch supports only one level in this release.

If traffic coming from a customer network is not tagged (native VLAN frames), these packets are bridged or routed as normal packets. All packets entering the service-provider network through a tunnel port on an edge switch are treated as untagged packets, whether they are untagged or already tagged with IEEE 802.1Q headers. The packets are encapsulated with the metro tag VLAN ID (set to the access VLAN of the tunnel port) when they are sent through the service-provider network on an IEEE 802.1Q trunk port. The priority field on the metro tag is set to the interface class of service (CoS) priority configured on the tunnel port. (The default is zero if none is configured.)

Configuring IEEE 802.1Q Tunneling

These sections contain this configuration information:

- Default IEEE 802.1Q Tunneling Configuration, page 13-4
- IEEE 802.1Q Tunneling Configuration Guidelines, page 13-4
- IEEE 802.1Q Tunneling and Other Features, page 13-6
- Configuring an IEEE 802.1Q Tunneling Port, page 13-6

Default IEEE 802.1Q Tunneling Configuration

By default, IEEE 802.1Q tunneling is disabled because the default switchport mode is access. Tagging of IEEE 802.1Q native VLAN packets on all IEEE 802.1Q trunk ports is also disabled. By default, VLANs on the switch are UNI isolated VLANs.

IEEE 802.1Q Tunneling Configuration Guidelines

When you configure IEEE 802.1Q tunneling, you should always use an asymmetrical link between the customer device and the edge switch, with the customer device port configured as an IEEE 802.1Q trunk port and the edge switch port configured as a tunnel port.

Assign tunnel ports only to VLANs that are used for tunneling.

Configuration requirements for native VLANs and for and maximum transmission units (MTUs) are explained in these next sections.

Native VLANs

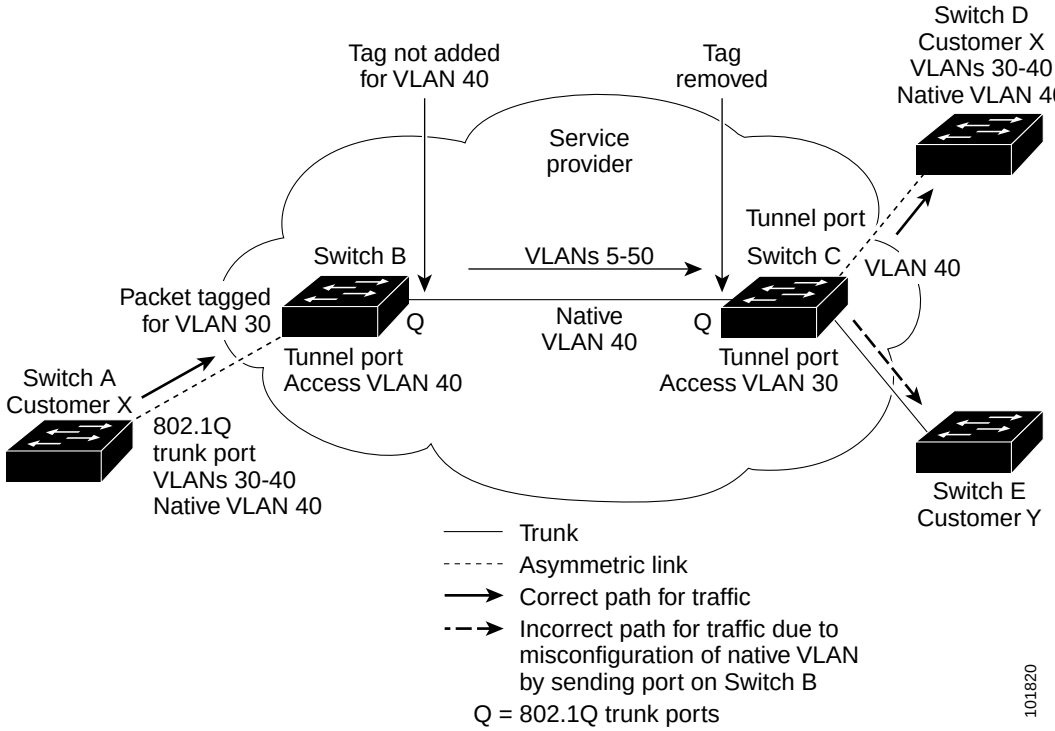
When configuring IEEE 802.1Q tunneling on an edge switch, you must use IEEE 802.1Q trunk ports for sending packets into the service-provider network. However, packets going through the core of the service-provider network can be carried through IEEE 802.1Q trunks, ISL trunks, or nontrunking links. When IEEE 802.1Q trunks are used in these core switches, the native VLANs of the IEEE 802.1Q trunks must not match any native VLAN of the nontrunking (tunneling) port on the same switch because traffic on the native VLAN would not be tagged on the IEEE 802.1Q sending trunk port.

See Figure 13-3. VLAN 40 is configured as the native VLAN for the IEEE 802.1Q trunk port from Customer X at the ingress edge switch in the service-provider network (Switch B). Switch A of Customer X sends a tagged packet on VLAN 30 to the ingress tunnel port of Switch B in the service-provider network, which belongs to access VLAN 40. Because the access VLAN of the tunnel port (VLAN 40) is the same as the native VLAN of the edge-switch trunk port (VLAN 40), the metro tag is not added to tagged packets received from the tunnel port. The packet carries only the VLAN 30 tag through the service-provider network to the trunk port of the egress-edge switch (Switch C) and is misdirected through the egress switch tunnel port to Customer Y.

These are some ways to solve this problem:

- Use ISL trunks between core switches in the service-provider network. Although customer interfaces connected to edge switches must be IEEE 802.1Q trunks, we recommend using ISL trunks for connecting switches in the core layer. The Cisco ME switch does not support ISL trunks.
- Use the **vlan dot1q tag native** global configuration command to configure the edge switch so that all packets going out an IEEE 802.1Q trunk, including the native VLAN, are tagged. If the switch is configured to tag native VLAN packets on all IEEE 802.1Q trunks, the switch accepts untagged packets, but sends only tagged packets.
- Ensure that the native VLAN ID on the edge-switch trunk port is not within the customer VLAN range. For example, if the trunk port carries traffic of VLANs 100 to 200, assign the native VLAN a number outside that range.

Figure 13-3 Potential Problem with IEEE 802.1Q Tunneling and Native VLANs



System MTU

The default system MTU for traffic on the switch is 1500 bytes. You can configure Fast Ethernet ports to support frames larger than 1500 bytes by using the **system mtu** global configuration command. You can configure Gigabit Ethernet ports to support frames larger than 1500 bytes by using the **system mtu jumbo** global configuration command. Because the IEEE 802.1Q tunneling feature increases the frame size by 4 bytes when the metro tag is added, you must configure all switches in the service-provider network to be able to process maximum frames by increasing the switch system MTU size to at least 1504 bytes. The maximum allowable system MTU for Gigabit Ethernet interfaces is 9000 bytes; the maximum system MTU for Fast Ethernet interfaces is 1546 bytes.

IEEE 802.1Q Tunneling and Other Features

Although IEEE 802.1Q tunneling works well for Layer 2 packet switching, there are incompatibilities between some Layer 2 features and Layer 3 switching.

Note Layer 3 switching is supported only when the metro IP access image is running on the switch.

- A tunnel port cannot be a routed port.
- IP routing is not supported on a VLAN that includes IEEE 802.1Q tunnel ports. Packets received from a tunnel port are forwarded based only on Layer 2 information. If routing is enabled on a switch virtual interface (SVI) that includes tunnel ports, untagged IP packets received from the tunnel port are recognized and routed by the switch. Customers can access the internet through its native VLAN. If this access is not needed, you should not configure SVIs on VLANs that include tunnel ports.
- Tunnel ports do not support IP access control lists (ACLs).
- Layer 3 quality of service (QoS) ACLs and other QoS features related to Layer 3 information are not supported on tunnel ports. MAC-based QoS is supported on tunnel ports.
- EtherChannel port groups are compatible with tunnel ports as long as the IEEE 802.1Q configuration is consistent within an EtherChannel port group.
- UniDirectional Link Detection (UDLD) is supported on IEEE 802.1Q tunnel ports.
- Port Aggregation Protocol (PAgP) and Link Aggregation Control Protocol (LACP) are supported only on IEEE 802.1Q tunnel ports that are network node interfaces (NNIs). UNIs do not support PAgP and LACP.
- Loopback detection is supported on IEEE 802.1Q tunnel ports.
- When an NNI port is configured as an IEEE 802.1Q tunnel port, spanning-tree bridge protocol data unit (BPDU) filtering is automatically enabled on the interface, and Cisco Discovery Protocol (CDP) is automatically disabled on the interface. UNIs do not support BPDU filtering or CDP.
- In a UNI isolated VLAN, IEEE 802.1Q tunneled access ports are isolated from each other, but in a UNI community VLAN, local switching occurs between these ports. For more information about UNI VLANs, see Chapter 11, “Configuring VLANs.”

Configuring an IEEE 802.1Q Tunneling Port

Beginning in privileged EXEC mode, follow these steps to configure a port as an IEEE 802.1Q tunnel port:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode for the interface to be configured as a tunnel port. This should be the edge port in the service-provider network that connects to the customer switch. Valid interfaces include physical interfaces and port-channel logical interfaces (port channels 1 to 48).
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled, and NNIs are enabled.

	Command	Purpose
Step 4	**switchport access vlan** *vlan-id*	Specify the default VLAN, which is used if the interface stops trunking. This VLAN ID is specific to the particular customer. **Note** If the VLAN is a UNI isolated VLAN, local switching does not occur between UNIs on the switch. If the VLAN is a UNI community VLAN, local switching is allowed.
Step 5	**switchport mode dot1q-tunnel**	Set the interface as an IEEE 802.1Q tunnel port.
Step 6	**exit**	Return to global configuration mode.
Step 7	**vlan dot1q tag native**	(Optional) Set the switch to enable tagging of native VLAN packets on all IEEE 802.1Q trunk ports. When not set, and a customer VLAN ID is the same as the native VLAN, the trunk port does not apply a metro tag, and packets could be sent to the wrong destination.
Step 8	**end**	Return to privileged EXEC mode.
Step 9	**show running-config**	Display the ports configured for IEEE 802.1Q tunneling.
	show dot1q-tunnel	Display the ports that are in tunnel mode.
Step 10	**show vlan dot1q tag native**	Display IEEE 802.1Q native VLAN tagging status.
Step 11	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no switchport mode dot1q-tunnel** interface configuration command to return the port to the default state of access. Use the **no vlan dot1q tag native** global configuration command to disable tagging of native VLAN packets.

This example shows how to configure an interface as a tunnel port, enable tagging of native VLAN packets, and verify the configuration. In this configuration, the VLAN ID for the customer connected to Gigabit Ethernet interface 2 is VLAN 22. This VLAN is by default a UNI isolated VLAN.

```
Switch(config)# interface gigabitethernet0/2
Switch(config-if)# switchport access vlan 22
% Access VLAN does not exist. Creating vlan 22
Switch(config-if)# switchport mode dot1q-tunnel
Switch(config-if)# exit
Switch(config)# vlan dot1q tag native
Switch(config)# end
Switch# show dot1q-tunnel interface gigabitethernet0/2
dot1q-tunnel mode LAN Port(s)
------------------------------
Gi0/1

Switch# show vlan dot1q tag native
dot1q native vlan tagging is enabled
```

Understanding Layer 2 Protocol Tunneling

Customers at different sites connected across a service-provider network need to use various Layer 2 protocols to scale their topologies to include all remote sites, as well as the local sites. STP must run properly, and every VLAN should build a proper spanning tree that includes the local site and all remote sites across the service-provider network. Cisco Discovery Protocol (CDP) must discover neighboring Cisco devices from local and remote sites. VLAN Trunking Protocol (VTP) must provide consistent VLAN configuration throughout all sites in the customer network that are participating in VTP.

Note The Cisco ME 3400 switch does not support VTP; CDP and STP are supported only on NNIs. However, Layer 2 protocol tunneling is supported on all ports on the switch.

When protocol tunneling is enabled, edge switches on the inbound side of the service-provider network encapsulate Layer 2 protocol packets with a special MAC address and send them across the service-provider network. Core switches in the network do not process these packets but forward them as normal packets. Layer 2 protocol data units (PDUs) for CDP, STP, or VTP cross the service-provider network and are delivered to customer switches on the outbound side of the service-provider network. Identical packets are received by all customer ports on the same VLANs with these results:

- Users on each of a customer's sites can properly run STP, and every VLAN can build a correct spanning tree based on parameters from all sites and not just from the local site.
- CDP discovers and shows information about the other Cisco devices connected through the service-provider network.
- VTP provides consistent VLAN configuration throughout the customer network, propagating to all switches through the service provider that support VTP.

Layer 2 protocol tunneling can be used independently or can enhance IEEE 802.1Q tunneling. If protocol tunneling is not enabled on IEEE 802.1Q tunneling ports, remote switches at the receiving end of the service-provider network do not receive the PDUs and cannot properly run STP, CDP, and VTP. When protocol tunneling *is* enabled, Layer 2 protocols within each customer's network are totally separate from those running within the service-provider network. Customer switches on different sites that send traffic through the service-provider network with IEEE 802.1Q tunneling achieve complete knowledge of the customer's VLAN. If IEEE 802.1Q tunneling is not used, you can still enable Layer 2 protocol tunneling by connecting to the customer switch through access ports and by enabling tunneling on the service-provider access port.

For example, in Figure 13-4, Customer X has four switches in the same VLAN, that are connected through the service-provider network. If the network does not tunnel PDUs, switches on the far ends of the network cannot properly run STP, CDP, and VTP. For example, STP for a VLAN on a switch in Customer X, Site 1, will build a spanning tree on the switches at that site without considering convergence parameters based on Customer X's switch in Site 2. This could result in the topology shown in Figure 13-5.

Figure 13-4 Layer 2 Protocol Tunneling

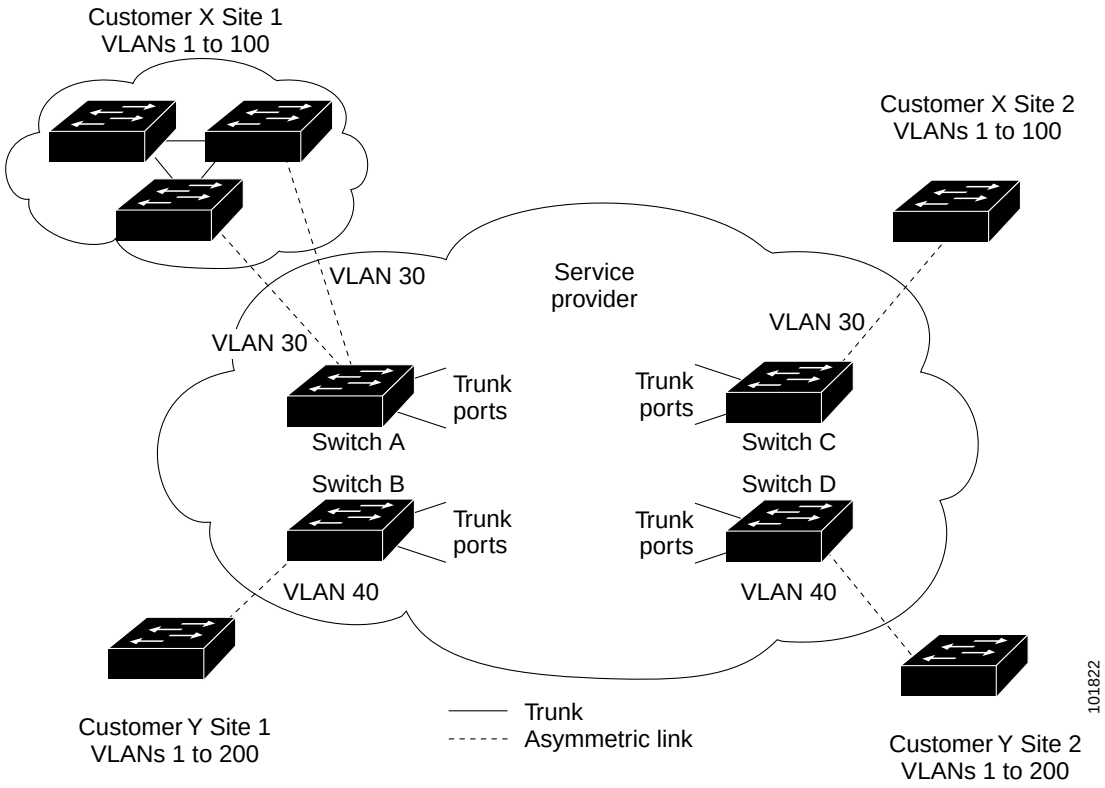
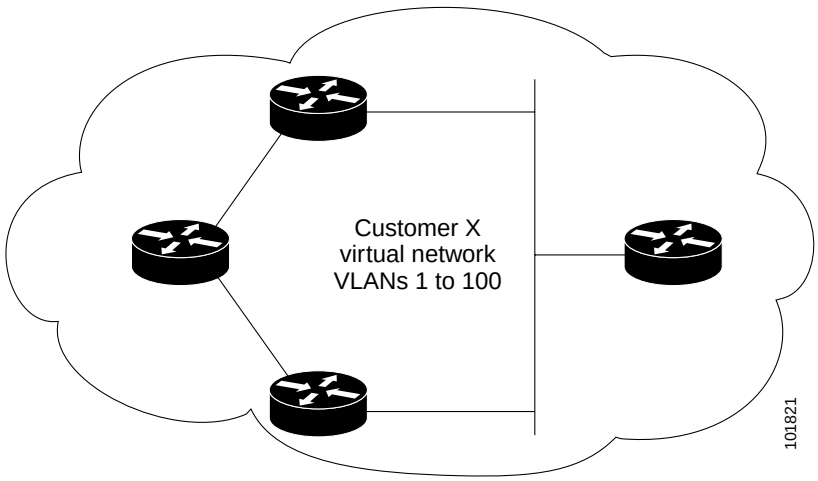


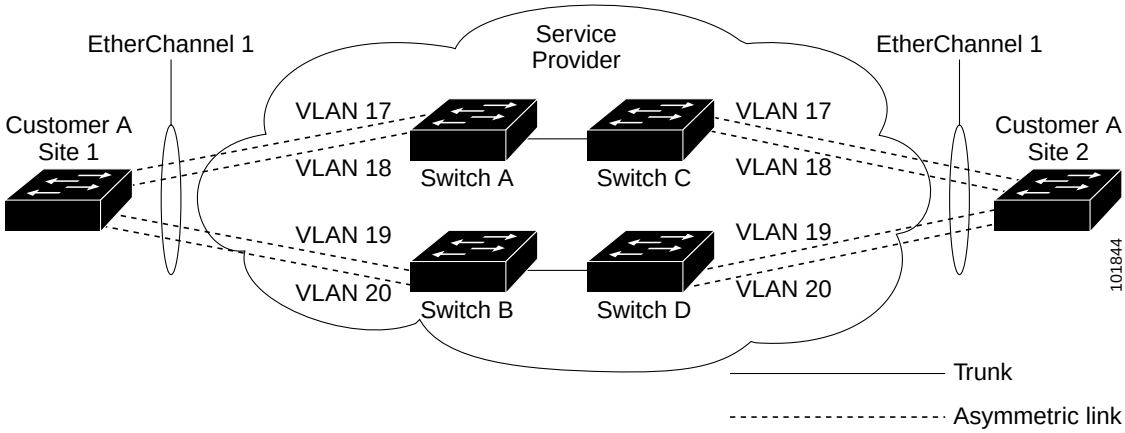
Figure 13-5 Layer 2 Network Topology without Proper Convergence



In an SP network, you can use Layer 2 protocol tunneling to enhance the creation of EtherChannels by emulating a point-to-point network topology. When you enable protocol tunneling (PAgP or LACP) on the SP switch, remote customer switches receive the PDUs and can negotiate the automatic creation of EtherChannels.

For example, in Figure 13-6, Customer A has two switches in the same VLAN that are connected through the SP network. When the network tunnels PDUs, switches on the far ends of the network can negotiate the automatic creation of EtherChannels without needing dedicated lines. See the "Configuring Layer 2 Tunneling for EtherChannels" section on page 13-14 for instructions.

Figure 13-6 Layer 2 Protocol Tunneling for EtherChannels



Configuring Layer 2 Protocol Tunneling

You can enable Layer 2 protocol tunneling (by protocol) on the ports that are connected to the customer in the edge switches of the service-provider network. The service-provider edge switches connected to the customer switch perform the tunneling process. Edge-switch tunnel ports are connected to customer IEEE 802.1Q trunk ports. Edge-switch access ports are connected to customer access ports. The edge switches connected to the customer switch perform the tunneling process.

You can enable Layer 2 protocol tunneling on ports that are configured as access ports or tunnel ports.

The switch supports Layer 2 protocol tunneling for CDP, STP, and VTP. For emulated point-to-point network topologies, it also supports PAgP, LACP, and UDLD protocols.

Caution PAgP, LACP, and UDLD protocol tunneling is only intended to emulate a point-to-point topology. An erroneous configuration that sends tunneled packets to many ports could lead to a network failure.

When the Layer 2 PDUs that entered the service-provider inbound edge switch through a Layer 2 protocol-enabled port exit through the trunk port into the service-provider network, the switch overwrites the customer PDU-destination MAC address with a well-known Cisco proprietary multicast address (01-00-0c-cd-cd-d0). If IEEE 802.1Q tunneling is enabled, packets are also double-tagged; the outer tag is the customer metro tag, and the inner tag is the customer's VLAN tag. The core switches ignore the inner tags and forward the packet to all trunk ports in the same metro VLAN. The edge switches on the outbound side restore the proper Layer 2 protocol and MAC address information and forward the packets to all tunnel or access ports in the same metro VLAN. Therefore, the Layer 2 PDUs remain intact and are delivered across the service-provider infrastructure to the other side of the customer network.

See Figure 13-4, with Customer X and Customer Y in access VLANs 30 and 40, respectively. Asymmetric links connect the customers in Site 1 to edge switches in the service-provider network. The Layer 2 PDUs (for example, BPDUs) coming into Switch 2 from Customer Y in Site 1 are forwarded to the infrastructure as double-tagged packets with the well-known MAC address as the destination MAC

address. These double-tagged packets have the metro VLAN tag of 40, as well as an inner VLAN tag (for example, VLAN 100). When the double-tagged packets enter Switch D, the outer VLAN tag 40 is removed, the well-known MAC address is replaced with the respective Layer 2 protocol MAC address, and the packet is sent to Customer Y on Site 2 as a single-tagged frame in VLAN 100.

You can also enable Layer 2 protocol tunneling on access ports on the edge switch connected to access or trunk ports on the customer switch. In this case, the encapsulation and decapsulation process is the same as described in the previous paragraph, except that the packets are not double-tagged in the service-provider network. The single tag is the customer-specific access VLAN tag.

These sections contain this configuration information:

- Default Layer 2 Protocol Tunneling Configuration, page 13-11
- Layer 2 Protocol Tunneling Configuration Guidelines, page 13-11
- Configuring Layer 2 Protocol Tunneling, page 13-12
- Configuring Layer 2 Tunneling for EtherChannels, page 13-14

Default Layer 2 Protocol Tunneling Configuration

Table 13-1 shows the default Layer 2 protocol tunneling configuration.

Table 13-1 Default Layer 2 Ethernet Interface VLAN Configuration

Feature	Default Setting
Layer 2 protocol tunneling	Disabled.
Shutdown threshold	None set.
Drop threshold	None set.
CoS value	If a CoS value is configured on the interface for data packets, that value is the default used for Layer 2 PDUs. If none is configured, the default is 5.

Layer 2 Protocol Tunneling Configuration Guidelines

These are some configuration guidelines and operating characteristics of Layer 2 protocol tunneling:

- The switch supports tunneling of CDP, STP, including multiple STP (MSTP), and VTP. Protocol tunneling is disabled by default but can be enabled for the individual protocols on IEEE 802.1Q tunnel ports or access ports.
- The edge switches on the outbound side of the service-provider network restore the proper Layer 2 protocol and MAC address information and forward the packets to all tunnel and access ports in the same metro VLAN.
- For interoperability with third-party vendor switches, the switch supports a Layer 2 protocol-tunnel bypass feature. Bypass mode transparently forwards control PDUs to vendor switches that have different ways of controlling protocol tunneling.When Layer 2 protocol tunneling is enabled on ingress ports on a switch, egress trunk ports forward the tunneled packets with a special encapsulation. If you also enable Layer 2 protocol tunneling on the egress trunk port, this behavior is bypassed, and the switch forwards control PDUs without any processing or modification.

- The switch supports PAgP, LACP, and UDLD tunneling for emulated point-to-point network topologies. Protocol tunneling is disabled by default but can be enabled for the individual protocols on IEEE 802.1Q tunnel ports or on access ports.
- If you enable PAgP or LACP tunneling, we recommend that you also enable UDLD on the interface for faster link-failure detection.
- Loopback detection is not supported on Layer 2 protocol tunneling of PAgP, LACP, or UDLD packets.
- EtherChannel port groups are compatible with tunnel ports when the IEEE 802.1Q configuration is consistent within an EtherChannel port group.
- If an encapsulated PDU (with the proprietary destination MAC address) is received from a tunnel port or an access port with Layer 2 tunneling enabled, the tunnel port is shut down to prevent loops. The port also shuts down when a configured shutdown threshold for the protocol is reached. You can manually re-enable the port (by entering a **shutdown** and a **no shutdown** command sequence). If errdisable recovery is enabled, the operation is retried after a specified time interval.
- Only decapsulated PDUs are forwarded to the customer network. The spanning-tree instance running on the service-provider network does not forward BPDUs to tunnel ports. CDP packets are not forwarded from tunnel ports.
- When protocol tunneling is enabled on an interface, you can set a per-protocol, per-port, shutdown threshold for the PDUs generated by the customer network. If the limit is exceeded, the port shuts down. You can also limit BPDU rate by using QoS ACLs and policy maps on a tunnel port.
- When protocol tunneling is enabled on an interface, you can set a per-protocol, per-port, drop threshold for the PDUs generated by the customer network. If the limit is exceeded, the port drops PDUs until the rate at which it receives them is below the drop threshold.
- Because tunneled PDUs (especially STP BPDUs) must be delivered to all remote sites so that the customer virtual network operates properly, you can give PDUs higher priority within the service-provider network than data packets received from the same tunnel port. By default, the PDUs use the same CoS value as data packets.

Configuring Layer 2 Protocol Tunneling

Beginning in privileged EXEC mode, follow these steps to configure a port for Layer 2 protocol tunneling:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and enter the interface to be configured as a tunnel port. This should be the edge port in the service-provider network that connects to the customer switch. Valid interfaces can be physical interfaces and port-channel logical interfaces (port channels 1 to 48).
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled, and NNIs are enabled.
Step 4	**switchport mode access** or **switchport mode dot1q-tunnel**	Configure the interface as an access port or an IEEE 802.1Q tunnel port. The default switchport mode is access.

	Command	Purpose
Step 5	**l2protocol-tunnel** [**cdp** \| **stp** \| **vtp**]	Enable protocol tunneling for the desired protocol. If no keyword is entered, tunneling is enabled for all three Layer 2 protocols.
Step 6	**l2protocol-tunnel shutdown-threshold** [**cdp** \| **stp** \| **vtp**] *value*	(Optional) Configure the threshold for packets-per-second accepted for encapsulation. The interface is disabled if the configured threshold is exceeded. If no protocol option is specified, the threshold applies to each of the tunneled Layer 2 protocol types. The range is 1 to 4096. The default is to have no threshold configured. **Note** If you also set a drop threshold on this interface, the shutdown-threshold value must be greater than or equal to the drop-threshold value.
Step 7	**l2protocol-tunnel drop-threshold** [**cdp** \| **stp** \| **vtp**] *value*	(Optional) Configure the threshold for packets-per-second accepted for encapsulation. The interface drops packets if the configured threshold is exceeded. If no protocol option is specified, the threshold applies to each of the tunneled Layer 2 protocol types. The range is 1 to 4096. The default is to have no threshold configured. If you also set a shutdown threshold on this interface, the drop-threshold value must be less than or equal to the shutdown-threshold value.
Step 8	**exit**	Return to global configuration mode.
Step 9	**errdisable recovery cause l2ptguard**	(Optional) Configure the recovery mechanism from a Layer 2 maximum-rate error so that the interface is re-enabled and can try again. Errdisable recovery is disabled by default; when enabled, the default time interval is 300 seconds.
Step 10	**l2protocol-tunnel cos** *value*	(Optional) Configure the CoS value for all tunneled Layer 2 PDUs. The range is 0 to 7; the default is the default CoS value for the interface. If none is configured, the default is 5.
Step 11	**end**	Return to privileged EXEC mode.
Step 12	**show l2protocol**	Display the Layer 2 tunnel ports on the switch, including the protocols configured, the thresholds, and the counters.
Step 13	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no l2protocol-tunnel** [**cdp** | **stp** | **vtp**] interface configuration command to disable protocol tunneling for one of the Layer 2 protocols or for all three. Use the **no l2protocol-tunnel shutdown-threshold** [**cdp** | **stp** | **vtp**] and the **no l2protocol-tunnel drop-threshold** [**cdp** | **stp** | **vtp**] commands to return the shutdown and drop thresholds to the default settings.

This example shows how to configure Layer 2 protocol tunneling for CDP, STP, and VTP and to verify the configuration.

```
Switch(config)# interface gigatethernet0/1
Switch(config-if)# l2protocol-tunnel cdp
Switch(config-if)# l2protocol-tunnel stp
Switch(config-if)# l2protocol-tunnel vtp
Switch(config-if)# l2protocol-tunnel shutdown-threshold 1500
Switch(config-if)# l2protocol-tunnel drop-threshold 1000
Switch(config-if)# exit
Switch(config)# l2protocol-tunnel cos 7
Switch(config)# end
```

```
Switch# show l2protocol
COS for Encapsulated Packets: 7
Port      Protocol Shutdown  Drop      Encapsulation Decapsulation Drop
                   Threshold Threshold Counter       Counter       Counter
-------   -------- --------- --------- ------------- ------------- -------------
Gi  0/1  cdp           1500      1000 2288          2282          0
          stp           1500      1000 116           13            0
          vtp           1500      1000 3             67            0
          pagp          ----      ---- 0             0             0
          lacp          ----      ---- 0             0             0
          udld          ----      ---- 0             0             0
```

Configuring Layer 2 Tunneling for EtherChannels

To configure Layer 2 point-to-point tunneling to facilitate the creation of EtherChannels, you need to configure both the SP edge switch and the customer switch.

Configuring the SP Edge Switch

Beginning in privileged EXEC mode, follow these steps to configure a SP edge switch for Layer 2 protocol tunneling for EtherChannels:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and enter the interface to be configured as a tunnel port. This should be the edge port in the SP network that connects to the customer switch. Valid interfaces are physical interfaces.
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled, and NNIs are enabled.
Step 4	**switchport mode dot1q-tunnel**	Configure the interface as an IEEE 802.1Q tunnel port.
Step 5	**l2protocol-tunnel point-to-point** [**pagp** \| **lacp** \| **udld**]	(Optional) Enable point-to-point protocol tunneling for the desired protocol. If no keyword is entered, tunneling is enabled for all three protocols. ⚠ **Caution** To avoid a network failure, make sure that the network is a point-to-point topology before you enable tunneling for PAgP, LACP, or UDLD packets.
Step 6	**l2protocol-tunnel shutdown-threshold** [**point-to-point** [**pagp** \| **lacp** \| **udld**]] *value*	(Optional) Configure the threshold for packets-per-second accepted for encapsulation. The interface is disabled if the configured threshold is exceeded. If no protocol option is specified, the threshold applies to each of the tunneled Layer 2 protocol types. The range is 1 to 4096. The default is to have no threshold configured. **Note** If you also set a drop threshold on this interface, the shutdown-threshold value must be greater than or equal to the drop-threshold value.

	Command	Purpose
Step 7	**l2protocol-tunnel drop-threshold** [**point-to-point** [**pagp** \| **lacp** \| **udld**]] *value*	(Optional) Configure the threshold for packets-per-second accepted for encapsulation. The interface drops packets if the configured threshold is exceeded. If no protocol option is specified, the threshold applies to each of the tunneled Layer 2 protocol types. The range is 1 to 4096. The default is to have no threshold configured. **Note** If you also set a shutdown threshold on this interface, the drop-threshold value must be less than or equal to the shutdown-threshold value.
Step 8	**no cdp enable**	If the interface is an NNI, disable CDP on the interface. UNIs do not support CDP.
Step 9	**spanning-tree bpdufilter enable**	If the interface is an NNI, enable BPDU filtering on the interface. UNIs do not support STP PBDU filtering.
Step 10	**exit**	Return to global configuration mode.
Step 11	**errdisable recovery cause l2ptguard**	(Optional) Configure the recovery mechanism from a Layer 2 maximum-rate error so that the interface is re-enabled and can try again. Errdisable recovery is disabled by default; when enabled, the default time interval is 300 seconds.
Step 12	**l2protocol-tunnel cos** *value*	(Optional) Configure the CoS value for all tunneled Layer 2 PDUs. The range is 0 to 7; the default is the default CoS value for the interface. If none is configured, the default is 5.
Step 13	**end**	Return to privileged EXEC mode.
Step 14	**show l2protocol**	Display the Layer 2 tunnel ports on the switch, including the protocols configured, the thresholds, and the counters.
Step 15	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no l2protocol-tunnel** [**point-to-point** [**pagp** | **lacp** | **udld**]] interface configuration command to disable point-to-point protocol tunneling for one of the Layer 2 protocols or for all three. Use the **no l2protocol-tunnel shutdown-threshold** [**point-to-point** [**pagp** | **lacp** | **udld**]] and the **no l2protocol-tunnel drop-threshold** [[**point-to-point** [**pagp** | **lacp** | **udld**]] commands to return the shutdown and drop thresholds to the default settings.

Configuring the Customer Switch

After configuring the SP edge switch, begin in privileged EXEC mode and follow these steps to configure a customer switch for Layer 2 protocol tunneling for EtherChannels:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter the interface configuration mode. This should be the customer switch port.
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled and NNIs are enabled.
Step 4	**switchport mode trunk**	Enable trunking on the interface.
Step 5	**udld enable**	Enable UDLD in **normal** mode on the interface.

	Command	Purpose
Step 6	**channel-group** *channel-group-number* **mode desirable**	Assign the interface to a channel group, and specify **desirable** for the PAgP mode if the interface is an NNI. For more information about configuring EtherChannels, see Chapter 31, "Configuring EtherChannels."
Step 7	**exit**	Return to global configuration mode.
Step 8	**interface port-channel** *port-channel number*	Enter port-channel interface mode.
Step 9	**shutdown**	Shut down the interface.
Step 10	**no shutdown**	Enable the interface.
Step 11	**end**	Return to privileged EXEC mode.
Step 12	**show l2protocol**	Display the Layer 2 tunnel ports on the switch, including the protocols configured, the thresholds, and the counters.
Step 13	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no switchport mode trunk**, the **no udld enable**, and the **no channel group** *channel-group-number* **mode desirable** interface configuration commands to return the interface to the default settings.

For EtherChannels, you need to configure both the SP edge switches and the customer switches for Layer 2 protocol tunneling. (See Figure 13-6 on page 13-10.)

This example shows how to configure the SP edge switch 1 and edge switch 2. VLANs 17, 18, 19, and 20 are the access VLANs, Gigabit Ethernet interfaces 1 and 2 are point-to-point tunnel ports with PAgP and UDLD enabled, the drop threshold is 1000, and Fast Ethernet interface 3 is a trunk port.

SP edge switch 1 configuration:

```
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# switchport access vlan 17
Switch(config-if)# switchport mode dot1q-tunnel
Switch(config-if)# l2protocol-tunnel point-to-point pagp
Switch(config-if)# l2protocol-tunnel point-to-point udld
Switch(config-if)# l2protocol-tunnel drop-threshold point-to-point pagp 1000
Switch(config-if)# exit
Switch(config)# interface gigabitethernet0/2
Switch(config-if)# switchport access vlan 18
Switch(config-if)# switchport mode dot1q-tunnel
Switch(config-if)# l2protocol-tunnel point-to-point pagp
Switch(config-if)# l2protocol-tunnel point-to-point udld
Switch(config-if)# l2protocol-tunnel drop-threshold point-to-point pagp 1000
Switch(config-if)# exit
Switch(config)# interface fastethernet0/3
Switch(config-if)# no shutdown
Switch(config-if)# switchport mode trunk
```

SP edge switch 2 configuration:

```
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# switchport access vlan 19
Switch(config-if)# switchport mode dot1q-tunnel
Switch(config-if)# l2protocol-tunnel point-to-point pagp
Switch(config-if)# l2protocol-tunnel point-to-point udld
Switch(config-if)# l2protocol-tunnel drop-threshold point-to-point pagp 1000
Switch(config-if)# exit
```

```
Switch(config)# interface gigabitethernet0/2
Switch(config-if)# switchport access vlan 20
Switch(config-if)# switchport mode dot1q-tunnel
Switch(config-if)# l2protocol-tunnel point-to-point pagp
Switch(config-if)# l2protocol-tunnel point-to-point udld
Switch(config-if)# l2protocol-tunnel drop-threshold point-to-point pagp 1000
Switch(config-if)# exit
Switch(config)# interface fastethernet0/3
Switch(config-if)# no shutdown
Switch(config-if)# switchport mode trunk
```

This example shows how to configure the customer switch at Site 1. Fast Ethernet interfaces 1, 2, 3, and 4 are set for IEEE 802.1Q trunking, UDLD is enabled, EtherChannel group 1 is enabled, and the port channel is shut down and then enabled to activate the EtherChannel configuration.

```
Switch(config)# interface fastethernet0/1
Switch(config-if)# no shutdown
Switch(config-if)# switchport mode trunk
Switch(config-if)# udld enable
Switch(config-if)# channel-group 1 mode desirable
Switch(config-if)# exit
Switch(config)# interface fastethernet0/2
Switch(config-if)# no shutdown
Switch(config-if)# switchport mode trunk
Switch(config-if)# udld enable
Switch(config-if)# channel-group 1 mode desirable
Switch(config-if)# exit
Switch(config)# interface fastethernet0/3
Switch(config-if)# no shutdown
Switch(config-if)# switchport mode trunk
Switch(config-if)# udld enable
Switch(config-if)# channel-group 1 mode desirable
Switch(config-if)# exit
Switch(config)# interface fastethernet0/4
Switch(config-if)# no shutdown
Switch(config-if)# switchport mode trunk
Switch(config-if)# udld enable
Switch(config-if)# channel-group 1 mode desirable
Switch(config-if)# exit
Switch(config)# interface port-channel 1
Switch(config-if)# shutdown
Switch(config-if)# no shutdown
Switch(config-if)# exit
```

Monitoring and Maintaining Tunneling Status

Table 13-2 shows the privileged EXEC commands for monitoring and maintaining IEEE 802.1Q and Layer 2 protocol tunneling.

Table 13-2 Commands for Monitoring and Maintaining Tunneling

Command	Purpose
clear l2protocol-tunnel counters	Clear the protocol counters on Layer 2 protocol tunneling ports.
show dot1q-tunnel	Display IEEE 802.1Q tunnel ports on the switch.
show dot1q-tunnel interface *interface-id*	Verify if a specific interface is a tunnel port.
show l2protocol-tunnel	Display information about Layer 2 protocol tunneling ports.
show errdisable recovery	Verify if the recovery timer from a Layer 2 protocol-tunnel error disable state is enabled.
show l2protocol-tunnel interface *interface-id*	Display information about a specific Layer 2 protocol tunneling port.
show l2protocol-tunnel summary	Display only Layer 2 protocol summary information.
show vlan dot1q tag native	Display the status of native VLAN tagging on the switch.

For detailed information about these displays, see the command reference for this release.



CHAPTER 14

Configuring STP

This chapter describes how to configure the Spanning Tree Protocol (STP) on port-based VLANs on the Cisco ME 3400 Ethernet Access switch. The switch uses the per-VLAN spanning-tree plus (PVST+) protocol based on the IEEE 802.1D standard and Cisco proprietary extensions, or it can use the rapid per-VLAN spanning-tree plus (rapid-PVST+) protocol based on the IEEE 802.1w standard. On the Cisco ME switch, STP is supported only on network node interfaces (NNIs). User network interfaces (UNIs) on the switch do not participate in STP and immediately forward traffic when they are brought up.

For information about the Multiple Spanning Tree Protocol (MSTP) and how to map multiple VLANs to the same spanning-tree instance, see Chapter 15, "Configuring MSTP." For information about other spanning-tree features such as Port Fast, root guard, and so forth, see Chapter 16, "Configuring Optional Spanning-Tree Features."

Note For complete syntax and usage information for the commands used in this chapter, see the command reference for this release.

This chapter consists of these sections:

- Understanding Spanning-Tree Features, page 14-1
- Configuring Spanning-Tree Features, page 14-11
- Displaying the Spanning-Tree Status, page 14-22

Understanding Spanning-Tree Features

These sections contain this conceptual information:

- STP Overview, page 14-2
- Spanning-Tree Topology and BPDUs, page 14-3
- Bridge ID, Switch Priority, and Extended System ID, page 14-4
- Spanning-Tree Interface States, page 14-4
- How a Switch or Port Becomes the Root Switch or Root Port, page 14-7
- Spanning Tree and Redundant Connectivity, page 14-8
- Spanning-Tree Address Management, page 14-8
- Accelerated Aging to Retain Connectivity, page 14-9

- Spanning-Tree Modes and Protocols, page 14-9
- Supported Spanning-Tree Instances, page 14-10
- Spanning-Tree Interoperability and Backward Compatibility, page 14-10
- STP and IEEE 802.1Q Trunks, page 14-10

For configuration information, see the "Configuring Spanning-Tree Features" section on page 14-11.

For information about optional spanning-tree features, see Chapter 16, "Configuring Optional Spanning-Tree Features."

STP Overview

STP is a Layer 2 link management protocol that provides path redundancy while preventing loops in the network. For a Layer 2 Ethernet network to function properly, only one active path can exist between any two stations. Multiple active paths among end stations cause loops in the network. If a loop exists in the network, end stations might receive duplicate messages. Switches might also learn end-station MAC addresses on multiple Layer 2 interfaces. These conditions result in an unstable network. Spanning-tree operation is transparent to end stations, which cannot detect whether they are connected to a single LAN segment or a switched LAN of multiple segments.

The STP uses a spanning-tree algorithm to select one switch of a redundantly connected network as the root of the spanning tree. The algorithm calculates the best loop-free path through a switched Layer 2 network by assigning a role to each port based on the role of the port in the active topology:

- Root—A forwarding port elected for the spanning-tree topology
- Designated—A forwarding port elected for every switched LAN segment
- Alternate—A blocked port providing an alternate path to the root bridge in the spanning tree
- Backup—A blocked port in a loopback configuration

Note On the Cisco ME 3400 switch, only NNIs participate in STP. Active UNIs are always in the forwarding state. In this overview, STP ports can be any interfaces on other switches, but only NNIs on a Cisco ME switch.

The switch that has *all* of its ports as the designated role or the backup role is the root switch. The switch that has at least *one* of its ports in the designated role is called the designated switch.

Spanning tree forces redundant data paths into a standby (blocked) state. If a network segment in the spanning tree fails and a redundant path exists, the spanning-tree algorithm recalculates the spanning-tree topology and activates the standby path. Switches send and receive spanning-tree frames, called bridge protocol data units (BPDUs), at regular intervals. The switches do not forward these frames but use them to construct a loop-free path. BPDUs contain information about the sending switch and its ports, including switch and MAC addresses, switch priority, port priority, and path cost. Spanning tree uses this information to elect the root switch and root port for the switched network and the root port and designated port for each switched segment.

When two ports on a switch are part of a loop, the spanning-tree port priority and path cost settings control which port is put in the forwarding state and which is put in the blocking state. The spanning-tree port priority value represents the location of a port in the network topology and how well it is located to pass traffic. The path cost value represents the media speed.

Note The switch sends keepalive messages (to ensure the connection is up) only on interfaces that do not have small form-factor pluggable (SFP) modules.

Spanning-Tree Topology and BPDUs

The stable, active spanning-tree topology of a switched network is controlled by these elements:

- The unique bridge ID (switch priority and MAC address) associated with each VLAN on each switch.
- The spanning-tree path cost to the root switch.
- The port identifier (port priority and MAC address) associated with each Layer 2 interface (or with each Layer 2 NNI in the Cisco ME switch).

When the switches in a network are powered up, each functions as the root switch. Each switch sends a configuration BPDU through all of its ports, or on the Cisco ME switch, only through the NNIs. The BPDUs communicate and compute the spanning-tree topology. Each configuration BPDU contains this information:

- The unique bridge ID of the switch that the sending switch identifies as the root switch
- The spanning-tree path cost to the root
- The bridge ID of the sending switch
- Message age
- The identifier of the sending interface
- Values for the hello, forward delay, and max-age protocol timers

When a switch receives a configuration BPDU that contains *superior* information (lower bridge ID, lower path cost, and so forth), it stores the information for that port. If this BPDU is received on the root port of the switch, the switch also forwards it with an updated message to all attached LANs for which it is the designated switch.

If a switch receives a configuration BPDU that contains *inferior* information to that currently stored for that port, it discards the BPDU. If the switch is a designated switch for the LAN from which the inferior BPDU was received, it sends that LAN a BPDU containing the up-to-date information stored for that port. In this way, inferior information is discarded, and superior information is propagated on the network.

A BPDU exchange results in these actions:

- One switch in the network is elected as the root switch (the logical center of the spanning-tree topology in a switched network).

 For each VLAN, the switch with the highest switch priority (the lowest numerical priority value) is elected as the root switch. If all switches are configured with the default priority (32768), the switch with the lowest MAC address in the VLAN becomes the root switch. The switch priority value occupies the most significant bits of the bridge ID, as shown in Table 14-1 on page 14-4.
- A root port is selected for each switch (except the root switch). On the Cisco ME switch, this port is always an NNI. This port provides the best path (lowest cost) when the switch forwards packets to the root switch.
- The shortest distance to the root switch is calculated for each switch based on the path cost.

- A designated switch for each LAN segment is selected. The designated switch incurs the lowest path cost when forwarding packets from that LAN to the root switch. The port through which the designated switch is attached to the LAN is called the designated port. For the Cisco ME switch, this only applies to NNIs.

All paths that are not needed to reach the root switch from anywhere in the switched network are placed in the spanning-tree blocking mode.

Bridge ID, Switch Priority, and Extended System ID

The IEEE 802.1D standard requires that each switch has an unique bridge identifier (bridge ID), which controls the selection of the root switch. Because each VLAN is considered as a different *logical bridge* with PVST+ and rapid PVST+, the same switch must have as many different bridge IDs as VLANs configured on it. Each VLAN on the switch has a unique 8-byte bridge ID. The two most-significant bytes are used for the switch priority, and the remaining six bytes are derived from the switch MAC address.

The switch supports the IEEE 802.1t spanning-tree extensions, and some of the bits previously used for the switch priority are now used as the VLAN identifier. The result is that fewer MAC addresses are reserved for the switch, and a larger range of VLAN IDs can be supported, all while maintaining the uniqueness of the bridge ID. As shown in Table 14-1, the two bytes previously used for the switch priority are reallocated into a 4-bit priority value and a 12-bit extended system ID value equal to the VLAN ID.

Table 14-1 Switch Priority Value and Extended System ID

Switch Priority Value				Extended System ID (Set Equal to the VLAN ID)											
Bit 16	**Bit 15**	**Bit 14**	**Bit 13**	**Bit 12**	**Bit 11**	**Bit 10**	**Bit 9**	**Bit 8**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**
32768	16384	8192	4096	2048	1024	512	256	128	64	32	16	8	4	2	1

Spanning tree uses the extended system ID, the switch priority, and the allocated spanning-tree MAC address to make the bridge ID unique for each VLAN.

Support for the extended system ID affects how you manually configure the root switch, the secondary root switch, and the switch priority of a VLAN. For example, when you change the switch priority value, you change the probability that the switch will be elected as the root switch. Configuring a higher value decreases the probability; a lower value increases the probability. For more information, see the "Configuring the Root Switch" section on page 14-14, the "Configuring a Secondary Root Switch" section on page 14-16, and the "Configuring the Switch Priority of a VLAN" section on page 14-19.

Spanning-Tree Interface States

Propagation delays can occur when protocol information passes through a switched LAN. As a result, topology changes can take place at different times and at different places in a switched network. When a port (NNI) transitions directly from nonparticipation in the spanning-tree topology to the forwarding state, it can create temporary data loops. Interfaces must wait for new topology information to propagate through the switched LAN before starting to forward frames. They must allow the frame lifetime to expire for forwarded frames that have used the old topology.

Each Layer 2 interface on a switch using spanning tree (or on a Cisco ME switch, each Layer 2 NNI) exists in one of these states:

- Blocking—The interface does not participate in frame forwarding.
- Listening—The first transitional state after the blocking state when the spanning tree determines that the interface should participate in frame forwarding.
- Learning—The interface prepares to participate in frame forwarding.
- Forwarding—The interface forwards frames.
- Disabled—The interface is not participating in spanning tree because of a shutdown port, no link on the port, or no spanning-tree instance running on the port.

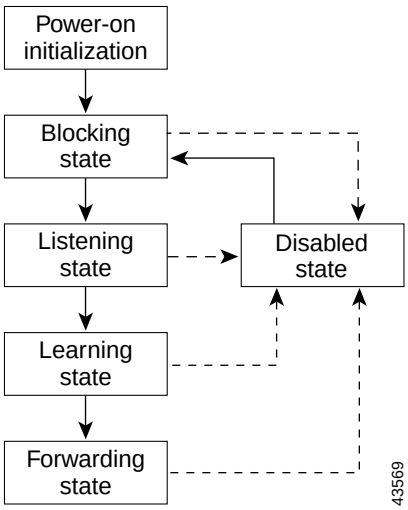
Note On a Cisco ME switch, UNIs are always in the forwarding state.

A port participating in spanning tree moves through these states:

- From initialization to blocking
- From blocking to listening or to disabled
- From listening to learning or to disabled
- From learning to forwarding or to disabled
- From forwarding to disabled

Figure 14-1 illustrates how an interface moves through the states.

Figure 14-1 Spanning-Tree Interface States



When you power up the switch, spanning tree is enabled by default, and every NNI in the Cisco ME switch, as well as any other port in other switches in the VLAN or network that are participating in spanning tree, goes through the blocking state and the transitory states of listening and learning. Spanning tree stabilizes each interface at the forwarding or blocking state.

Note UNIs are shut down by default, and when they are brought up, they immediately start forwarding traffic.

When the spanning-tree algorithm places a Layer 2 spanning-tree interface in the forwarding state, this process occurs:

1. The interface is in the listening state while spanning tree waits for protocol information to transition the interface to the blocking state.
2. While spanning tree waits the forward-delay timer to expire, it moves the interface to the learning state and resets the forward-delay timer.
3. In the learning state, the interface continues to block frame forwarding as the switch learns end-station location information for the forwarding database.
4. When the forward-delay timer expires, spanning tree moves the interface to the forwarding state, where both learning and frame forwarding are enabled.

Blocking State

A Layer 2 interface in the blocking state does not participate in frame forwarding. After initialization, a BPDU is sent to each switch interface, or to each switch NNI. A switch initially functions as the root until it exchanges BPDUs with other switches. This exchange establishes which switch in the network is the root or root switch. If there is only one switch in the network, no exchange occurs, the forward-delay timer expires, and the interface moves to the listening state. An interface participating in spanning tree always enters the blocking state after switch initialization.

An interface in the blocking state performs these functions:

- Discards frames received on the interface
- Discards frames switched from another interface for forwarding
- Does not learn addresses
- Receives BPDUs

Listening State

The listening state is the first state a Layer 2 interface enters after the blocking state. The interface enters this state when the spanning tree decides that the interface should participate in frame forwarding.

An interface in the listening state performs these functions:

- Discards frames received on the interface
- Discards frames switched from another interface for forwarding
- Does not learn addresses
- Receives BPDUs

Learning State

A Layer 2 interface in the learning state prepares to participate in frame forwarding. The interface enters the learning state from the listening state.

An interface in the learning state performs these functions:

- Discards frames received on the interface
- Discards frames switched from another interface for forwarding
- Learns addresses
- Receives BPDUs

Forwarding State

A Layer 2 interface in the forwarding state forwards frames. The interface enters the forwarding state from the learning state.

An interface in the forwarding state performs these functions:

- Receives and forwards frames received on the interface
- Forwards frames switched from another interface
- Learns addresses
- Receives BPDUs

Disabled State

A Layer 2 interface in the disabled state does not participate in frame forwarding or in the spanning tree. An interface in the disabled state is nonoperational.

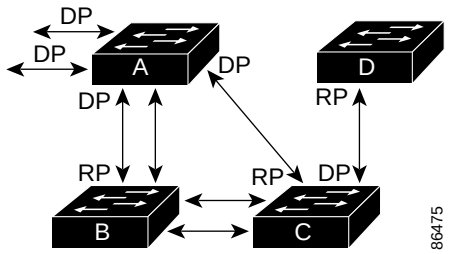
A disabled interface performs these functions:

- Discards frames received on the interface
- Discards frames switched from another interface for forwarding
- Does not learn addresses
- Does not receive BPDUs

How a Switch or Port Becomes the Root Switch or Root Port

If all switches in a network are enabled with default spanning-tree settings, the switch with the lowest MAC address becomes the root switch. In Figure 14-2, Switch A is elected as the root switch because the switch priority of all the switches is set to the default (32768) and Switch A has the lowest MAC address. However, because of traffic patterns, number of forwarding interfaces, or link types, Switch A might not be the ideal root switch. By increasing the priority (lowering the numerical value) of the ideal switch so that it becomes the root switch, you force a spanning-tree recalculation to form a new topology with the ideal switch as the root.

Figure 14-2 Spanning-Tree Topology



RP = Root Port
DP = Designated Port

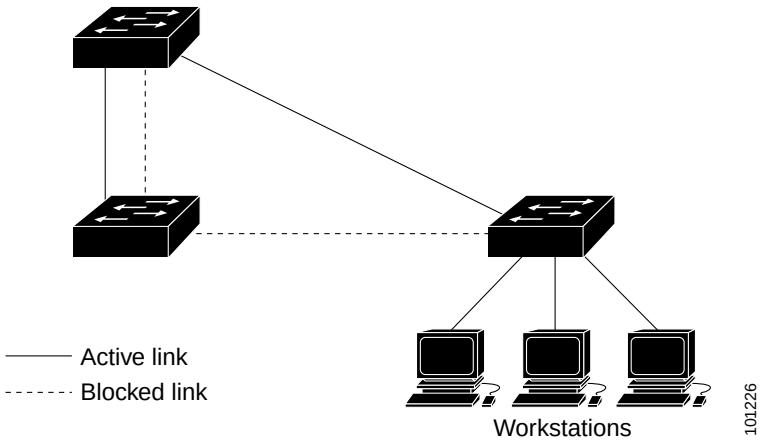
When the spanning-tree topology is calculated based on default parameters, the path between source and destination end stations in a switched network might not be ideal. For instance, connecting higher-speed links to an interface that has a higher number than the root port can cause a root-port change. The goal is to make the fastest link the root port.

For example, assume that one port on Switch B is a Gigabit Ethernet link and that another port on Switch B (a 10/100 link) is the root port. Network traffic might be more efficient over the Gigabit Ethernet link. By changing the spanning-tree port priority on the Gigabit Ethernet port to a higher priority (lower numerical value) than the root port, the Gigabit Ethernet port becomes the new root port.

Spanning Tree and Redundant Connectivity

You can create a redundant backbone with spanning tree by connecting two switch interfaces that are participating in spanning tree to another device or to two different devices, as shown in Figure 14-3. Spanning tree automatically disables one interface but enables it if the other one fails. If one link is high-speed and the other is low-speed, the low-speed link is always disabled. If the speeds are the same, the port priority and port ID are added together, and spanning tree disables the link with the lowest value.

Figure 14-3 Spanning Tree and Redundant Connectivity



You can also create redundant links between switches by using EtherChannel groups. For more information, see Chapter 31, “Configuring EtherChannels.”

Spanning-Tree Address Management

IEEE 802.1D specifies 17 multicast addresses, ranging from 0x00180C2000000 to 0x0180C2000010, to be used by different bridge protocols. These addresses are static addresses that cannot be removed.

Regardless of the spanning-tree state, each switch receives but does not forward packets destined for addresses between 0x0180C2000000 and 0x0180C200000F.

If spanning tree is enabled, the CPU on the switch receives packets destined for 0x0180C2000000 and 0x0180C2000010. If spanning tree is disabled, the switch forwards those packets as unknown multicast addresses.

Accelerated Aging to Retain Connectivity

The default for aging dynamic addresses is 5 minutes, the default setting of the **mac address-table aging-time** global configuration command. However, a spanning-tree reconfiguration can cause many station locations to change. Because these stations could be unreachable for 5 minutes or more during a reconfiguration, the address-aging time is accelerated so that station addresses can be dropped from the address table and then relearned. The accelerated aging is the same as the forward-delay parameter value (**spanning-tree vlan** *vlan-id* **forward-time** *seconds* global configuration command) when the spanning tree reconfigures.

Because each VLAN is a separate spanning-tree instance, the switch accelerates aging on a per-VLAN basis. A spanning-tree reconfiguration on one VLAN can cause the dynamic addresses learned on that VLAN to be subject to accelerated aging. Dynamic addresses on other VLANs can be unaffected and remain subject to the aging interval entered for the switch.

Spanning-Tree Modes and Protocols

The NNIs on the switch support these spanning-tree modes and protocols:

- PVST+—This spanning-tree mode is based on the IEEE 802.1D standard and Cisco proprietary extensions. It is the default spanning-tree mode used on most Ethernet port-based VLANs. The PVST+ runs on each VLAN on the switch up to the maximum supported, ensuring that each has a loop-free path through the network.

 The PVST+ provides Layer 2 load balancing for the VLAN on which it runs. You can create different logical topologies by using the VLANs on your network to ensure that all of your links are used but that no one link is oversubscribed. Each instance of PVST+ on a VLAN has a single root switch. This root switch propagates the spanning-tree information associated with that VLAN to all other switches in the network. Because each switch has the same information about the network, this process ensures that the network topology is maintained.

- Rapid PVST+—This spanning-tree mode is the same as PVST+ except that is uses a rapid convergence based on the IEEE 802.1w standard. This is the default spanning-tree mode for the Cisco ME switch NNIs. Rapid PVST+ is compatible with PVST+. To provide rapid convergence, the rapid PVST+ immediately deletes dynamically learned MAC address entries on a per-port basis upon receiving a topology change. By contrast, PVST+ uses a short aging time for dynamically learned MAC address entries.

 The rapid PVST+ uses the same configuration as PVST+ (except where noted), and the switch needs only minimal extra configuration. The benefit of rapid PVST+ is that you can migrate a large PVST+ install base to rapid PVST+ without having to learn the complexities of the MSTP configuration and without having to reprovision your network. In rapid-PVST+ mode, each VLAN runs its own spanning-tree instance up to the maximum supported.

- MSTP—This spanning-tree mode is based on the IEEE 802.1s standard. You can map multiple VLANs to the same spanning-tree instance, which reduces the number of spanning-tree instances required to support a large number of VLANs. The MSTP runs on top of the RSTP (based on IEEE 802.1w), which provides for rapid convergence of the spanning tree by eliminating the forward delay and by quickly transitioning root ports and designated ports to the forwarding state. You cannot run MSTP without RSTP.

 The most common initial deployment of MSTP is in the backbone and distribution layers of a Layer 2 switched network. For more information, see Chapter 15, “Configuring MSTP.”

For information about the number of supported spanning-tree instances, see the next section.

Supported Spanning-Tree Instances

In PVST+ or rapid-PVST+ mode, the switch supports up to 128 spanning-tree instances.

In MSTP mode, the switch supports up to 16 MST instances. The number of VLANs that can be mapped to a particular MST instance is unlimited.

Spanning-Tree Interoperability and Backward Compatibility

Table 14-2 lists the interoperability and compatibility among the supported spanning-tree modes in a network.

Table 14-2 PVST+, MSTP, and Rapid-PVST+ Interoperability

	PVST+	MSTP	Rapid PVST+
PVST+	Yes	Yes (with restrictions)	Yes (reverts to PVST+)
MSTP	Yes (with restrictions)	Yes	Yes (reverts to PVST+)
Rapid PVST+	Yes (reverts to PVST+)	Yes (reverts to PVST+)	Yes

In a mixed MSTP and PVST+ network, the common spanning-tree (CST) root must be inside the MST backbone, and a PVST+ switch cannot connect to multiple MST regions.

When a network contains switches running rapid PVST+ and switches running PVST+, we recommend that the rapid-PVST+ switches and PVST+ switches be configured for different spanning-tree instances. In the rapid-PVST+ spanning-tree instances, the root switch must be a rapid-PVST+ switch. In the PVST+ instances, the root switch must be a PVST+ switch. The PVST+ switches should be at the edge of the network.

STP and IEEE 802.1Q Trunks

The IEEE 802.1Q standard for VLAN trunks imposes some limitations on the spanning-tree strategy for a network. The standard requires only one spanning-tree instance for *all* VLANs allowed on the trunks. However, in a network of Cisco switches connected through IEEE 802.1Q trunks, the switches maintain one spanning-tree instance for *each* VLAN allowed on the trunks.

When you connect a Cisco switch to a non-Cisco device through an IEEE 802.1Q trunk, the Cisco switch uses PVST+ to provide spanning-tree interoperability. If rapid PVST+ is enabled, the switch uses it instead of PVST+. The switch combines the spanning-tree instance of the IEEE 802.1Q VLAN of the trunk with the spanning-tree instance of the non-Cisco 802.1Q switch.

However, all PVST+ or rapid-PVST+ information is maintained by Cisco switches separated by a cloud of non-Cisco 802.1Q switches. The non-Cisco 802.1Q cloud separating the Cisco switches is treated as a single trunk link between the switches.

PVST+ is automatically enabled on IEEE 802.1Q trunks, and no user configuration is required. The external spanning-tree behavior on access ports is not affected by PVST+.

For more information on IEEE 802.1Q trunks, see Chapter 11, “Configuring VLANs.”

Configuring Spanning-Tree Features

These sections contain this configuration information:

- Default Spanning-Tree Configuration, page 14-11
- Spanning-Tree Configuration Guidelines, page 14-12
- Changing the Spanning-Tree Mode., page 14-13 (required)
- Disabling Spanning Tree, page 14-14 (optional)
- Configuring the Root Switch, page 14-14 (optional)
- Configuring a Secondary Root Switch, page 14-16 (optional)
- Configuring Port Priority, page 14-16 (optional)
- Configuring Path Cost, page 14-18 (optional)
- Configuring the Switch Priority of a VLAN, page 14-19 (optional)
- Configuring Spanning-Tree Timers, page 14-20 (optional)

Default Spanning-Tree Configuration

Table 14-3 shows the default spanning-tree configuration.

Table 14-3 Default Spanning-Tree Configuration

Feature	Default Setting
Enable state	Enabled on NNIs in VLAN 1. For more information, see the "Supported Spanning-Tree Instances" section on page 14-10.
Spanning-tree mode	Rapid PVST+ Rapid PVST+ interoperates with PVST and PVST+. MSTP is disabled.
Switch priority	32768.
Spanning-tree port priority (configurable on a per-interface basis)	128.
Spanning-tree port cost (configurable on a per-interface basis)	1000 Mbps: 4. 100 Mbps: 19. 10 Mbps: 100.
Spanning-tree VLAN port priority (configurable on a per-VLAN basis)	128.
Spanning-tree VLAN port cost (configurable on a per-VLAN basis)	1000 Mbps: 4. 100 Mbps: 19. 10 Mbps: 100.
Spanning-tree timers	Hello time: 2 seconds. Forward-delay time: 15 seconds. Maximum-aging time: 20 seconds.

Spanning-Tree Configuration Guidelines

If more VLANs are defined than there are spanning-tree instances, you can enable PVST+ or rapid PVST+ on NNIs in only 128 VLANs on the switch. The remaining VLANs operate with spanning tree disabled. However, you can map multiple VLANs to the same spanning-tree instances by using MSTP. For more information, see Chapter 15, "Configuring MSTP."

If 128 instances of spanning tree are already in use, you can disable spanning tree on NNIs in one of the VLANs and then enable it on the VLAN where you want it to run. Use the **no spanning-tree vlan** *vlan-id* global configuration command to disable spanning tree on a specific VLAN, and use the **spanning-tree vlan** *vlan-id* global configuration command to enable spanning tree on the desired VLAN.

Caution Switches that are not running spanning tree still forward BPDUs that they receive so that the other switches on the VLAN that have a running spanning-tree instance can break loops. Therefore, spanning tree must be running on enough switches to break all the loops in the network; for example, at least one switch on each loop in the VLAN must be running spanning tree. It is not absolutely necessary to run spanning tree on all switches in the VLAN. However, if you are running spanning tree only on a minimal set of switches, an incautious change to the network that introduces another loop into the VLAN can result in a broadcast storm.

Note If you have already used all available spanning-tree instances on your switch, adding another VLAN creates a VLAN that is not running spanning tree on that switch. If you have the default allowed list on the trunk ports of that switch, the new VLAN is carried on all trunk ports. Depending on the topology of the network, this could create a loop in the new VLAN that will not be broken, particularly if there are several adjacent switches that have all run out of spanning-tree instances. You can prevent this possibility by setting up allowed lists on the trunk ports of switches that have used up their allocation of spanning-tree instances. Setting up allowed lists is not necessary in many cases and can make it more labor-intensive to add another VLAN to the network.

Spanning-tree commands control the configuration of VLAN spanning-tree instances. You create a spanning-tree instance when you assign an NNI to a VLAN. The spanning-tree instance is removed when the last NNI is moved to another VLAN. You can configure switch and port parameters before a spanning-tree instance is created; these parameters are applied when the spanning-tree instance is created.

The switch supports PVST+, rapid PVST+, and MSTP, but only one version can be active at any time. (For example, all VLANs run PVST+, all VLANs run rapid PVST+, or all VLANs run MSTP.) For information about the different spanning-tree modes and how they interoperate, see the "Spanning-Tree Interoperability and Backward Compatibility" section on page 14-10.

Changing the Spanning-Tree Mode.

The switch supports three spanning-tree modes: PVST+, rapid PVST+, or MSTP. By default, the switch runs the rapid PVST+ protocol on all NNIs.

Beginning in privileged EXEC mode, follow these steps to change the spanning-tree mode. If you want to enable a mode that is different from the default mode, this procedure is required.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree mode** {**pvst** \| **mst** \| **rapid-pvst**}	Configure a spanning-tree mode on NNIs on the switch. • Select **pvst** to enable PVST+. • Select **mst** to enable MSTP (and RSTP). For more configuration steps, see Chapter 15, "Configuring MSTP." • Select **rapid-pvst** to enable rapid PVST+ (the default setting).
Step 3	**interface** *interface-id*	(Recommended only for rapid-PVST+ mode) Specify an NNI to configure, and enter interface configuration mode. Valid interfaces include physical NNIs, VLANs, and NNI port channels. The VLAN ID range is 1 to 4094. The port-channel range is 1 to 12. **Note** If a physical interface is not an NNI, enter the **port-type nni** interface configuration command before attempting to configure it as a spanning-tree link. If the interface is a VLAN, only NNIs in the VLAN will run spanning tree. If the interface is a port channel, all members of the port channel must be NNIs.
Step 4	**spanning-tree link-type point-to-point**	(Recommended only for rapid-PVST+ mode) Specify that the link type for this port is point-to-point. If you connect this NNI (local port) to a remote port through a point-to-point link and the local port becomes a designated port, the switch negotiates with the remote port and rapidly changes the local port to the forwarding state.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**clear spanning-tree detected-protocols**	(Recommended only for rapid-PVST+ mode) If any NNI on the switch is connected to a port on a legacy IEEE 802.1D switch, restart the protocol migration process on the entire switch. This step is optional if the designated switch detects that this switch is running rapid PVST+.
Step 7	**show spanning-tree summary** and **show spanning-tree interface** *interface-id*	Verify your entries.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no spanning-tree mode** global configuration command. To return the NNI to its default spanning-tree setting, use the **no spanning-tree link-type** interface configuration command.

Disabling Spanning Tree

Spanning tree is enabled by default on all NNIs in VLAN 1 and in all newly created VLANs up to the spanning-tree limit specified in the "Supported Spanning-Tree Instances" section on page 14-10. Disable spanning tree only if you are sure there are no loops in the network topology.

Caution When spanning tree is disabled and loops are present in the topology, excessive traffic and indefinite packet duplication can drastically reduce network performance.

Beginning in privileged EXEC mode, follow these steps to disable spanning-tree on a per-VLAN basis. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**no spanning-tree vlan** *vlan-id*	For *vlan-id*, the range is 1 to 4094.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show spanning-tree vlan** *vlan-id*	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To re-enable spanning-tree, use the **spanning-tree vlan** *vlan-id* global configuration command.

Configuring the Root Switch

The switch maintains a separate spanning-tree instance for each active VLAN configured on it. A bridge ID, consisting of the switch priority and the switch MAC address, is associated with each instance. For each VLAN, the switch with the lowest bridge ID becomes the root switch for that VLAN.

To configure a switch to become the root for the specified VLAN, use the **spanning-tree vlan** *vlan-id* **root** global configuration command to modify the switch priority from the default value (32768) to a significantly lower value. When you enter this command, the software checks the switch priority of the root switches for each VLAN. Because of the extended system ID support, the switch sets its own priority for the specified VLAN to 24576 if this value will cause this switch to become the root for the specified VLAN.

If any root switch for the specified VLAN has a switch priority lower than 24576, the switch sets its own priority for the specified VLAN to 4096 less than the lowest switch priority. (4096 is the value of the least-significant bit of a 4-bit switch priority value as shown in Table 14-1 on page 14-4.)

Note The **spanning-tree vlan** *vlan-id* **root** global configuration command fails if the value necessary to be the root switch is less than 1.

Note If your network consists of switches that both do and do not support the extended system ID, it is unlikely that the switch with the extended system ID support will become the root switch. The extended system ID increases the switch priority value every time the VLAN number is greater than the priority of the connected switches running older software.

Note The root switch for each spanning-tree instance should be a backbone or distribution switch. Do not configure an access switch as the spanning-tree primary root.

Use the **diameter** keyword to specify the Layer 2 network diameter (that is, the maximum number of switch hops between any two end stations in the Layer 2 network). When you specify the network diameter, the switch automatically sets an optimal hello time, forward-delay time, and maximum-age time for a network of that diameter, which can significantly reduce the convergence time. You can use the **hello** keyword to override the automatically calculated hello time.

Note After configuring the switch as the root switch, we recommend that you avoid manually configuring the hello time, forward-delay time, and maximum-age time through the **spanning-tree vlan** *vlan-id* **hello-time**, **spanning-tree vlan** *vlan-id* **forward-time**, and the **spanning-tree vlan** *vlan-id* **max-age** global configuration commands.

Beginning in privileged EXEC mode, follow these steps to configure a switch to become the root for the specified VLAN. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree vlan** *vlan-id* **root primary** [**diameter** *net-diameter* [**hello-time** *seconds*]]	Configure a switch to become the root for the specified VLAN. • For *vlan-id*, you can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. • (Optional) For **diameter** *net-diameter*, specify the maximum number of switches between any two end stations. The range is 2 to 7. • (Optional) For **hello-time** *seconds*, specify the interval in seconds between the generation of configuration messages by the root switch. The range is 1 to 10; the default is 2.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show spanning-tree detail**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no spanning-tree vlan** *vlan-id* **root** global configuration command.

Configuring a Secondary Root Switch

When you configure a switch as the secondary root, the switch priority is modified from the default value (32768) to 28672. The switch is then likely to become the root switch for the specified VLAN if the primary root switch fails. This is assuming that the other network switches use the default switch priority of 32768 and therefore are unlikely to become the root switch.

You can execute this command on more than one switch to configure multiple backup root switches. Use the same network diameter and hello-time values that you used when you configured the primary root switch with the **spanning-tree vlan** *vlan-id* **root primary** global configuration command.

Beginning in privileged EXEC mode, follow these steps to configure a switch to become the secondary root for the specified VLAN. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree vlan** *vlan-id* **root secondary** [**diameter** *net-diameter* [**hello-time** *seconds*]]	Configure a switch to become the secondary root for the specified VLAN. • For *vlan-id*, you can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. • (Optional) For **diameter** *net-diameter*, specify the maximum number of switches between any two end stations. The range is 2 to 7. • (Optional) For **hello-time** *seconds*, specify the interval in seconds between the generation of configuration messages by the root switch. The range is 1 to 10; the default is 2. Use the same network diameter and hello-time values that you used when configuring the primary root switch. See the "Configuring the Root Switch" section on page 14-14.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show spanning-tree detail**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no spanning-tree vlan** *vlan-id* **root** global configuration command.

Configuring Port Priority

If a loop occurs, spanning tree uses the port priority when selecting an NNI to put into the forwarding state. You can assign higher priority values (lower numerical values) to NNIs that you want selected first and lower priority values (higher numerical values) to ones that you want selected last. If all NNIs have the same priority value, spanning tree puts the NNI with the lowest interface number in the forwarding state and blocks the other interfaces.

Beginning in privileged EXEC mode, follow these steps to configure the port priority of an NNI. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify an interface to configure, and enter interface configuration mode. Valid interfaces include physical NNIs and port-channel logical interfaces (**port-channel** *port-channel-number*) that contain only NNIs. If a physical interface is not an NNI, enter the **port-type nni** interface configuration command to make it an NNI before configuring spanning tree.
Step 3	**spanning-tree port-priority** *priority*	Configure the port priority for an NNI. For *priority*, the range is 0 to 240, in increments of 16; the default is 128. Valid values are 0, 16, 32, 48, 64, 80, 96, 112, 128, 144, 160, 176, 192, 208, 224, and 240. All other values are rejected. The lower the number, the higher the priority.
Step 4	**spanning-tree vlan** *vlan-id* **port-priority** *priority*	Configure the port priority for a VLAN. • For *vlan-id*, you can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. • For *priority*, the range is 0 to 240, in increments of 16; the default is 128. Valid values are 0, 16, 32, 48, 64, 80, 96, 112, 128, 144, 160, 176, 192, 208, 224, and 240. All other values are rejected. The lower the number, the higher the priority.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show spanning-tree interface** *interface-id* or **show spanning-tree vlan** *vlan-id*	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Note The **show spanning-tree interface** *interface-id* privileged EXEC command displays information only if the port is in a link-up operative state. Otherwise, you can use the **show running-config interface** privileged EXEC command to confirm the configuration.

To return to the default spanning-tree setting, use the **no spanning-tree** [**vlan** *vlan-id*] **port-priority** interface configuration command. For information on how to configure load sharing on trunk ports by using spanning-tree port priorities, see the "Configuring Trunk Ports for Load Sharing" section on page 11-19.

Configuring Path Cost

The spanning-tree path cost default value is derived from the media speed of an interface (NNI or port channel of multiple NNIs). If a loop occurs, spanning tree uses cost when selecting an interface to put in the forwarding state. You can assign lower cost values to interfaces that you want selected first and higher cost values that you want selected last. If all NNIs (or port channels) have the same cost value, spanning tree puts the interface with the lowest interface number in the forwarding state and blocks the other interfaces.

Beginning in privileged EXEC mode, follow these steps to configure the cost of an interface. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify an interface to configure, and enter interface configuration mode. Valid interfaces include physical NNIs and port-channel logical interfaces (**port-channel** *port-channel-number*) that contain only NNIs.
Step 3	**spanning-tree cost** *cost*	Configure the cost for an interface. If a loop occurs, spanning tree uses the path cost when selecting an interface to place into the forwarding state. A lower path cost represents higher-speed transmission. For *cost*, the range is 1 to 200000000; the default value is derived from the media speed of the interface.
Step 4	**spanning-tree vlan** *vlan-id* **cost** *cost*	Configure the cost for a VLAN. If a loop occurs, spanning tree uses the path cost when selecting an NNI to place into the forwarding state. A lower path cost represents higher-speed transmission. • For *vlan-id*, you can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. • For *cost*, the range is 1 to 200000000; the default value is derived from the media speed of the interface.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show spanning-tree interface** *interface-id* or **show spanning-tree vlan** *vlan-id*	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Note The **show spanning-tree interface** *interface-id* privileged EXEC command displays information only for ports that are in a link-up operative state. Otherwise, you can use the **show running-config** privileged EXEC command to confirm the configuration.

To return to the default setting, use the **no spanning-tree** [**vlan** *vlan-id*] **cost** interface configuration command. For information on how to configure load sharing on trunk ports by using spanning-tree path costs, see the "Configuring Trunk Ports for Load Sharing" section on page 11-19.

Configuring the Switch Priority of a VLAN

You can configure the switch priority and make it more likely that the switch will be chosen as the root switch.

Note Exercise care when using this command. For most situations, we recommend that you use the **spanning-tree vlan** *vlan-id* **root primary** and the **spanning-tree vlan** *vlan-id* **root secondary** global configuration commands to modify the switch priority.

Beginning in privileged EXEC mode, follow these steps to configure the switch priority of a VLAN. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree vlan** *vlan-id* **priority** *priority*	Configure the switch priority of a VLAN. • For *vlan-id*, you can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. • For *priority*, the range is 0 to 61440 in increments of 4096; the default is 32768. The lower the number, the more likely the switch will be chosen as the root switch. Valid priority values are 4096, 8192, 12288, 16384, 20480, 24576, 28672, 32768, 36864, 40960, 45056, 49152, 53248, 57344, and 61440. All other values are rejected.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show spanning-tree vlan** *vlan-id*	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no spanning-tree vlan** *vlan-id* **priority** global configuration command.

Configuring Spanning-Tree Timers

Table 14-4 describes the timers that affect the entire spanning-tree performance.

Table 14-4 Spanning-Tree Timers

Variable	Description
Hello timer	Controls how often the switch broadcasts hello messages to other switches.
Forward-delay timer	Controls how long each of the listening and learning states last before the NNI begins forwarding.
Maximum-age timer	Controls the amount of time the switch stores protocol information received on an NNI.

The sections that follow provide the configuration steps.

Configuring the Hello Time

You can configure the interval between the generation of configuration messages by the root switch by changing the hello time.

Note Exercise care when using this command. For most situations, we recommend that you use the **spanning-tree vlan** *vlan-id* **root primary** and the **spanning-tree vlan** *vlan-id* **root secondary** global configuration commands to modify the hello time.

Beginning in privileged EXEC mode, follow these steps to configure the hello time of a VLAN. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree vlan** *vlan-id* **hello-time** *seconds*	Configure the hello time of a VLAN. The hello time is the interval between the generation of configuration messages by the root switch. These messages mean that the switch is alive. • For *vlan-id*, you can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. • For *seconds*, the range is 1 to 10; the default is 2.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show spanning-tree vlan** *vlan-id*	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no spanning-tree vlan** *vlan-id* **hello-time** global configuration command.

Configuring the Forwarding-Delay Time for a VLAN

Beginning in privileged EXEC mode, follow these steps to configure the forwarding-delay time for a VLAN. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree vlan** *vlan-id* **forward-time** *seconds*	Configure the forward time of a VLAN. The forward delay is the number of seconds an NNI waits before changing from its spanning-tree learning and listening states to the forwarding state. • For *vlan-id*, you can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. • For *seconds*, the range is 4 to 30; the default is 15.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show spanning-tree vlan** *vlan-id*	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no spanning-tree vlan** *vlan-id* **forward-time** global configuration command.

Configuring the Maximum-Aging Time for a VLAN

Beginning in privileged EXEC mode, follow these steps to configure the maximum-aging time for a VLAN. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree vlan** *vlan-id* **max-age** *seconds*	Configure the maximum-aging time of a VLAN. The maximum-aging time is the number of seconds a switch waits without receiving spanning-tree configuration messages before attempting a reconfiguration. • For *vlan-id*, you can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. • For *seconds*, the range is 6 to 40; the default is 20.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show spanning-tree vlan** *vlan-id*	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no spanning-tree vlan** *vlan-id* **max-age** global configuration command.

Displaying the Spanning-Tree Status

To display the spanning-tree status, use one or more of the privileged EXEC commands in Table 14-5:

Table 14-5 Commands for Displaying Spanning-Tree Status

Command	Purpose
show spanning-tree active	Displays spanning-tree information only on active NNIs.
show spanning-tree detail	Displays a detailed summary of interface information.
show spanning-tree interface *interface-id*	Displays spanning-tree information for the specified NNI.
show spanning-tree summary [**totals**]	Displays a summary of interface states or displays the total lines of the STP state section.

You can clear spanning-tree counters by using the **clear spanning-tree** [**interface** *interface-id*] privileged EXEC command.

For information about other keywords for the **show spanning-tree** privileged EXEC command, see the command reference for this release.



CHAPTER 15

Configuring MSTP

This chapter describes how to configure the Cisco implementation of the IEEE 802.1s Multiple STP (MSTP) on the Cisco ME 3400 Ethernet Access switch. On the Cisco ME switch, STP is supported only on network node interfaces (NNIs). User network interfaces (UNIs) on the switch do not participate in STP and forward traffic immediately when they are brought up.

Note The multiple spanning-tree (MST) implementation is a pre-standard implementation. It is based on the draft version of the IEEE standard.

The MSTP enables multiple VLANs to be mapped to the same spanning-tree instance, thereby reducing the number of spanning-tree instances needed to support a large number of VLANs. The MSTP provides for multiple forwarding paths for data traffic and enables load balancing. It improves the fault tolerance of the network because a failure in one instance (forwarding path) does not affect other instances (forwarding paths). The most common initial deployment of MSTP is in the backbone and distribution layers of a Layer 2 switched network. This deployment provides the highly available network required in a service-provider environment.

When the switch is in the MST mode, the Rapid Spanning Tree Protocol (RSTP), which is based on IEEE 802.1w, is automatically enabled. The RSTP provides rapid convergence of the spanning tree through explicit handshaking that eliminates the IEEE 802.1D forwarding delay and quickly transitions root ports and designated ports to the forwarding state.

Both MSTP and RSTP improve the spanning-tree operation and maintain backward compatibility with equipment that is based on the (original) 802.1D spanning tree, with existing Cisco-proprietary Multiple Instance STP (MISTP), and with existing Cisco per-VLAN spanning-tree plus (PVST+) and rapid per-VLAN spanning-tree plus (rapid PVST+). For information about PVST+ and rapid PVST+, see Chapter 14, “Configuring STP.” For information about other spanning-tree features such as Port Fast, UplinkFast, root guard, and so forth, see Chapter 16, “Configuring Optional Spanning-Tree Features.”

Note For complete syntax and usage information for the commands used in this chapter, see the command reference for this release.

This chapter consists of these sections:

- Understanding MSTP, page 15-2
- Understanding RSTP, page 15-6
- Configuring MSTP Features, page 15-11
- Displaying the MST Configuration and Status, page 15-23

Understanding MSTP

MSTP, which uses RSTP for rapid convergence, enables VLANs to be grouped into a spanning-tree instance, with each instance having a spanning-tree topology independent of other spanning-tree instances. This architecture provides multiple forwarding paths for data traffic, enables load balancing, and reduces the number of spanning-tree instances required to support a large number of VLANs.

These sections describe how the MSTP works:

- Multiple Spanning-Tree Regions, page 15-2
- IST, CIST, and CST, page 15-2
- Hop Count, page 15-4
- Boundary Ports, page 15-5
- "Interoperability with IEEE 802.1D STP" section on page 15-5

For configuration information, see the "Configuring MSTP Features" section on page 15-11.

Multiple Spanning-Tree Regions

For switches to participate in multiple spanning-tree (MST) instances, you must consistently configure the switches with the same MST configuration information. A collection of interconnected switches that have the same MST configuration comprises an MST region as shown in Figure 15-1 on page 15-4.

The MST configuration controls to which MST region each switch belongs. The configuration includes the name of the region, the revision number, and the MST VLAN-to-instance assignment map. You configure the switch for a region by using the **spanning-tree mst configuration** global configuration command, after which the switch enters the MST configuration mode. From this mode, you can map VLANs to an MST instance by using the **instance** MST configuration command, specify the region name by using the **name** MST configuration command, and set the revision number by using the **revision** MST configuration command.

A region can have one member or multiple members with the same MST configuration; each member must be capable of processing RSTP bridge protocol data units (BPDUs). There is no limit to the number of MST regions in a network, but each region can support up to 16 spanning-tree instances. You can assign a VLAN to only one spanning-tree instance at a time.

IST, CIST, and CST

Unlike PVST+ and rapid PVST+ in which all the spanning-tree instances are independent, the MSTP establishes and maintains two types of spanning trees:

- An internal spanning tree (IST), which is the spanning tree that runs in an MST region.

 Within each MST region, the MSTP maintains multiple spanning-tree instances. Instance 0 is a special instance for a region, known as the internal spanning tree (IST). All other MST instances are numbered from 1 to 15.

 The IST is the only spanning-tree instance that sends and receives BPDUs; all of the other spanning-tree instance information is contained in M-records, which are encapsulated within MSTP BPDUs. Because the MSTP BPDU carries information for all instances, the number of BPDUs that need to be processed by a switch to support multiple spanning-tree instances is significantly reduced.

All MST instances within the same region share the same protocol timers, but each MST instance has its own topology parameters, such as root switch ID, root path cost, and so forth. By default, all VLANs are assigned to the IST.

An MST instance is local to the region; for example, MST instance 1 in region A is independent of MST instance 1 in region B, even if regions A and B are interconnected.

- A common and internal spanning tree (CIST), which is a collection of the ISTs in each MST region, and the common spanning tree (CST) that interconnects the MST regions and single spanning trees.

 The spanning tree computed in a region appears as a subtree in the CST that encompasses the entire switched domain. The CIST is formed as a result of the spanning-tree algorithm running between switches that support the IEEE 802.1w, IEEE 802.1s, and IEEE 802.1D protocols. The CIST inside an MST region is the same as the CST outside a region.

For more information, see the "Operations Within an MST Region" section on page 15-3 and the "Operations Between MST Regions" section on page 15-3.

Operations Within an MST Region

The IST connects all the MSTP switches in a region. When the IST converges, the root of the IST becomes the IST master (shown in Figure 15-1 on page 15-4), which is the switch within the region with the lowest bridge ID and path cost to the CST root. The IST master also is the CST root if there is only one region within the network. If the CST root is outside the region, one of the MSTP switches at the boundary of the region is selected as the IST master.

When an MSTP switch initializes, it sends BPDUs claiming itself as the root of the CST and the IST master, with both of the path costs to the CST root and to the IST master set to zero. The switch also initializes all of its MST instances and claims to be the root for all of them. If the switch receives superior MST root information (lower bridge ID, lower path cost, and so forth) than currently stored for the port, it relinquishes its claim as the IST master.

During initialization, a region might have many subregions, each with its own IST master. As switches receive superior IST information, they leave their old subregions and join the new subregion that might contain the true IST master. Thus all subregions shrink, except for the one that contains the true IST master.

For correct operation, all switches in the MST region must agree on the same IST master. Therefore, any two switches in the region synchronize their port roles for an MST instance only if they converge to a common IST master.

Operations Between MST Regions

If there are multiple regions or legacy IEEE 802.1D switches within the network, MSTP establishes and maintains the CST, which includes all MST regions and all legacy STP switches in the network. The MST instances combine with the IST at the boundary of the region to become the CST.

The IST connects all the MSTP switches in the region and appears as a subtree in the CST that encompasses the entire switched domain, with the root of the subtree being the IST master. The MST region appears as a virtual switch to adjacent STP switches and MST regions.

Figure 15-1 shows a network with three MST regions and a legacy IEEE 802.1D switch (D). The IST master for region 1 (A) is also the CST root. The IST master for region 2 (B) and the IST master for region 3 (C) are the roots for their respective subtrees within the CST. The RSTP runs in all regions.

Figure 15-1 MST Regions, IST Masters, and the CST Root

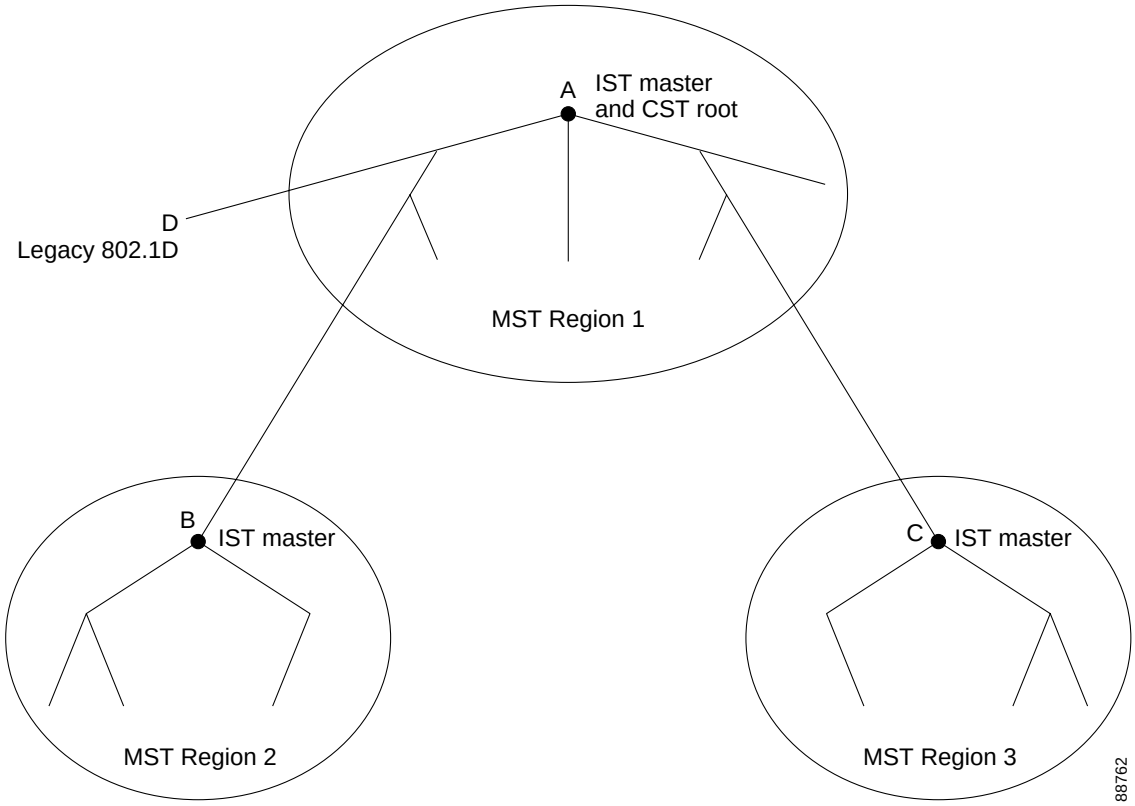


Figure 15-1 does not show additional MST instances for each region. Note that the topology of MST instances can be different from that of the IST for the same region.

Only the CST instance sends and receives BPDUs, and MST instances add their spanning-tree information into the BPDUs to interact with neighboring switches and compute the final spanning-tree topology. Because of this, the spanning-tree parameters related to BPDU transmission (for example, hello time, forward time, max-age, and max-hops) are configured only on the CST instance but affect all MST instances. Parameters related to the spanning-tree topology (for example, switch priority, port VLAN cost, port VLAN priority) can be configured on both the CST instance and the MST instance.

MSTP switches use Version 3 RSTP BPDUs or 802.1D STP BPDUs to communicate with legacy IEEE 802.1D switches. MSTP switches use MSTP BPDUs to communicate with MSTP switches.

Hop Count

The IST and MST instances do not use the message-age and maximum-age information in the configuration BPDU to compute the spanning-tree topology. Instead, they use the path cost to the root and a hop-count mechanism similar to the IP time-to-live (TTL) mechanism.

By using the **spanning-tree mst max-hops** global configuration command, you can configure the maximum hops inside the region and apply it to the IST and all MST instances in that region. The hop count achieves the same result as the message-age information (trigger a reconfiguration). The root switch of the instance always sends a BPDU (or M-record) with a cost of 0 and the hop count set to the

maximum value. When a switch receives this BPDU, it decrements the received remaining hop count by one and propagates this value as the remaining hop count in the BPDUs it generates. When the count reaches zero, the switch discards the BPDU and ages the information held for the port.

The message-age and maximum-age information in the RSTP portion of the BPDU remain the same throughout the region, and the same values are propagated by the region's designated ports at the boundary.

Boundary Ports

A boundary port is a port that connects an MST region to a single spanning-tree region running RSTP, to a single spanning-tree region running PVST+ or rapid PVST+, or to another MST region with a different MST configuration.A boundary port also connects to a LAN, the designated switch of which is either a single spanning-tree switch or a switch with a different MST configuration.



Note On the Cisco ME switch, only NNIs are MST ports. UNIs do not participate in STP.

At the boundary, the roles of the MST ports do not matter, and their state is forced to be the same as the IST port state (MST ports at the boundary are in the forwarding state only when the IST port is forwarding). An IST port at the boundary can have any port role except a backup port role.

On a shared boundary link, the MST ports wait in the blocking state for the forward-delay time to expire before transitioning to the learning state. The MST ports wait another forward-delay time before transitioning to the forwarding state.

If the boundary port is on a point-to-point link and it is the IST root port, the MST ports transition to the forwarding state as soon as the IST port transitions to the forwarding state.

If the IST port is a designated port on a point-to-point link and if the IST port transitions to the forwarding state because of an agreement received from its peer port, the MST ports also immediately transition to the forwarding state.

If a boundary port transitions to the forwarding state in an IST instance, it is forwarding in all MST instances, and a topology change is triggered. If a boundary port with the IST root or designated port role receives a topology change notice external to the MST cloud, the MSTP switch triggers a topology change in the IST instance and in all the MST instances active on that port.

Interoperability with IEEE 802.1D STP

A switch running MSTP supports a built-in protocol migration mechanism that enables it to interoperate with legacy IEEE 802.1D switches. If this switch receives a legacy IEEE 802.1D configuration BPDU (a BPDU with the protocol version set to 0), it sends only IEEE 802.1D BPDUs on that port. An MSTP switch also can detect that a port is at the boundary of a region when it receives a legacy BPDU, an MSTP BPDU (Version 3) associated with a different region, or an RSTP BPDU (Version 2).

However, the switch does not automatically revert to the MSTP mode if it no longer receives IEEE 802.1D BPDUs because it cannot detect whether the legacy switch has been removed from the link unless the legacy switch is the designated switch. Also, a switch might continue to assign a boundary role to a port when the switch to which this switch is connected has joined the region. To restart the protocol migration process (force the renegotiation with neighboring switches), use the **clear spanning-tree detected-protocols** privileged EXEC command.

If all the legacy switches on the link are RSTP switches, they can process MSTP BPDUs as if they are RSTP BPDUs. Therefore, MSTP switches send either a Version 0 configuration and TCN BPDUs or Version 3 MSTP BPDUs on a boundary port. A boundary port connects to a LAN, the designated switch of which is either a single spanning-tree switch or a switch with a different MST configuration.

Understanding RSTP

The RSTP takes advantage of point-to-point wiring and provides rapid convergence of the spanning tree. Reconfiguration of the spanning tree can occur in less than 1 second (in contrast to 50 seconds with the default settings in the IEEE 802.1D spanning tree), which is critical for networks carrying delay-sensitive traffic such as voice and video.

These section describes how the RSTP works:

- Port Roles and the Active Topology, page 15-6
- Rapid Convergence, page 15-7
- Synchronization of Port Roles, page 15-8
- Bridge Protocol Data Unit Format and Processing, page 15-9

For configuration information, see the "Configuring MSTP Features" section on page 15-11.

Port Roles and the Active Topology

The RSTP provides rapid convergence of the spanning tree by assigning port roles and by learning the active topology. The RSTP builds upon the IEEE 802.1D STP to select the switch with the highest switch priority (lowest numerical priority value) as the root switch as described in the "Spanning-Tree Topology and BPDUs" section on page 14-3. Then the RSTP assigns one of these port roles to individual ports.

Note On the Cisco ME switch, only NNIs are RSTP ports. UNIs do not participate in STP.

- Root port—Provides the best path (lowest cost) when the switch forwards packets to the root switch.
- Designated port—Connects to the designated switch, which incurs the lowest path cost when forwarding packets from that LAN to the root switch. The port through which the designated switch is attached to the LAN is called the designated port.
- Alternate port—Offers an alternate path toward the root switch to that provided by the current root port.
- Backup port—Acts as a backup for the path provided by a designated port toward the leaves of the spanning tree. A backup port can exist only when two ports are connected together in a loopback by a point-to-point link or when a switch has two or more connections to a shared LAN segment.
- Disabled port—Has no role within the operation of the spanning tree.

A port with the root or a designated port role is included in the active topology. A port with the alternate or backup port role is excluded from the active topology.

In a stable topology with consistent port roles throughout the network, the RSTP ensures that every root port and designated port immediately transition to the forwarding state while all alternate and backup ports are always in the discarding state (equivalent to blocking in 802.1D). The port state controls the operation of the forwarding and learning processes. Table 15-1 provides a comparison of IEEE 802.1D and RSTP port states.

Table 15-1 Port State Comparison

Operational Status	STP Port State (IEEE 802.1D)	RSTP Port State	Is Port Included in the Active Topology?
Enabled	Blocking	Discarding	No
Enabled	Listening	Discarding	No
Enabled	Learning	Learning	Yes
Enabled	Forwarding	Forwarding	Yes
Disabled	Disabled	Discarding	No

To be consistent with Cisco STP implementations, this guide documents the port state as *blocking* instead of *discarding*. Designated ports start in the listening state.

Rapid Convergence

The RSTP provides for rapid recovery of connectivity following the failure of a switch, a switch port, or a LAN. It provides rapid convergence for edge ports, new root ports, and ports connected through point-to-point links as follows:

- Edge ports—If you configure a port as an edge port on an RSTP switch by using the **spanning-tree portfast** interface configuration command, the edge port immediately transitions to the forwarding state. An edge port is the same as a Port Fast-enabled port, and you should enable it only on ports that connect to a single end station.
- Root ports—If the RSTP selects a new root port, it blocks the old root port and immediately transitions the new root port to the forwarding state.
- Point-to-point links—If you connect a port to another port through a point-to-point link and the local port becomes a designated port, it negotiates a rapid transition with the other port by using the proposal-agreement handshake to ensure a loop-free topology.

Note On the Cisco ME switch, these ports are always NNIs.

As shown in Figure 15-2, Switch A is connected to Switch B through a point-to-point link, and all of the ports are in the blocking state. Assume that the priority of Switch A is a smaller numerical value than the priority of Switch B. Switch A sends a proposal message (a configuration BPDU with the proposal flag set) to Switch B, proposing itself as the designated switch.

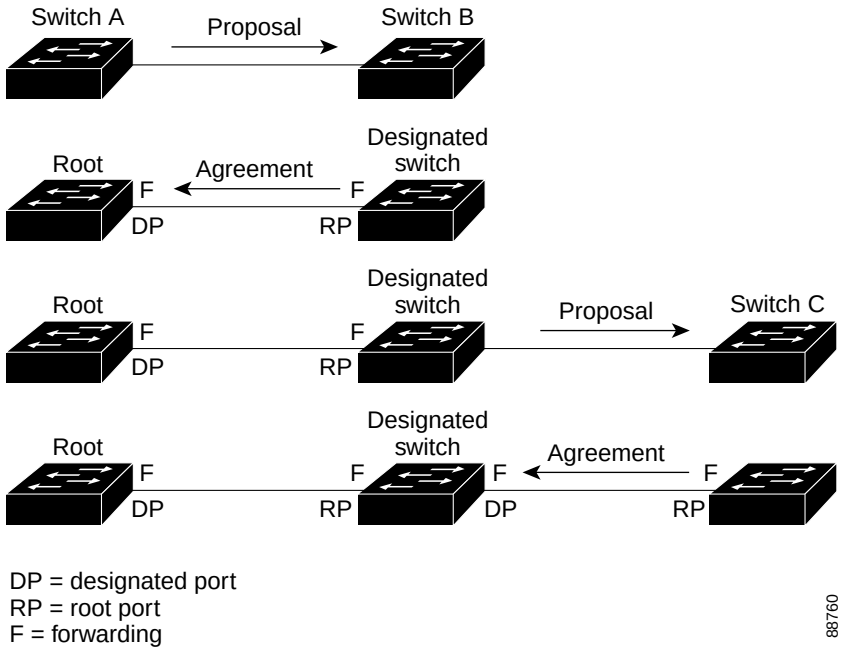
After receiving the proposal message, Switch B selects as its new root port the port from which the proposal message was received, forces all nonedge ports to the blocking state, and sends an agreement message (a BPDU with the agreement flag set) through its new root port.

After receiving Switch B's agreement message, Switch A also immediately transitions its designated port to the forwarding state. No loops in the network are formed because Switch B blocked all of its nonedge ports and because there is a point-to-point link between Switches A and B.

When Switch C is connected to Switch B, a similar set of handshaking messages are exchanged. Switch C selects the port connected to Switch B as its root port, and both ends immediately transition to the forwarding state. With each iteration of this handshaking process, one more switch joins the active topology. As the network converges, this proposal-agreement handshaking progresses from the root toward the leaves of the spanning tree.

The switch learns the link type from the port duplex mode: a full-duplex port is considered to have a point-to-point connection; a half-duplex port is considered to have a shared connection. You can override the default setting that is controlled by the duplex setting by using the **spanning-tree link-type** interface configuration command.

Figure 15-2 Proposal and Agreement Handshaking for Rapid Convergence



Synchronization of Port Roles

When the switch receives a proposal message on one of its ports and that port is selected as the new root port, the RSTP forces all other ports to synchronize with the new root information.

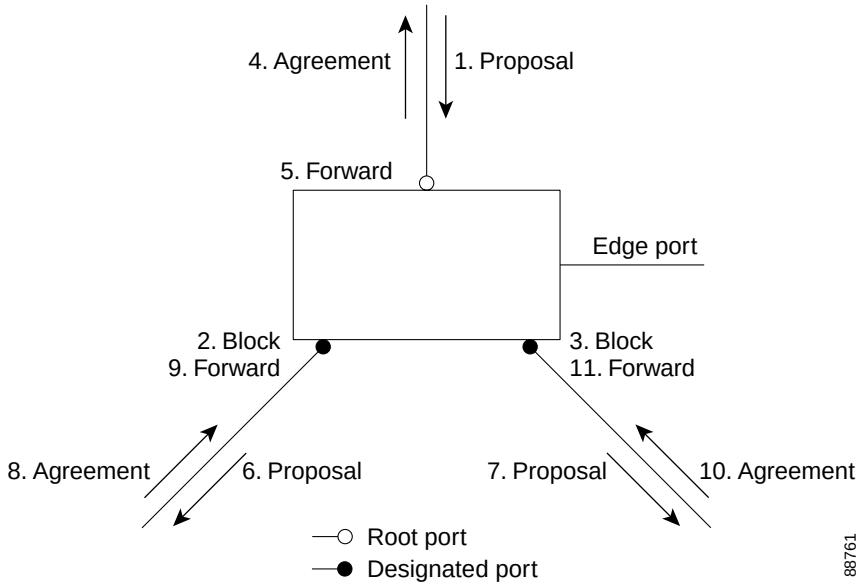
The switch is synchronized with superior root information received on the root port if all other ports are synchronized. An individual port on the switch is synchronized if

- That port is in the blocking state.
- It is an edge port (a port configured to be at the edge of the network).

If a designated port (NNI on the Cisco ME switch) is in the forwarding state and is not configured as an edge port, it transitions to the blocking state when the RSTP forces it to synchronize with new root information. In general, when the RSTP forces a port to synchronize with root information and the port does not satisfy any of the above conditions, its port state is set to blocking.

After ensuring all of the ports are synchronized, the switch sends an agreement message to the designated switch corresponding to its root port. When the switches connected by a point-to-point link are in agreement about their port roles, the RSTP immediately transitions the port states to forwarding. The sequence of events is shown in Figure 15-3.

Figure 15-3 Sequence of Events During Rapid Convergence



Bridge Protocol Data Unit Format and Processing

The RSTP BPDU format is the same as the IEEE 802.1D BPDU format except that the protocol version is set to 2. A new one-byte Version 1 Length field is set to zero, which means that no version 1 protocol information is present. Table 15-2 shows the RSTP flag fields.

Table 15-2 RSTP BPDU Flags

Bit	Function
0	Topology change (TC)
1	Proposal
2–3:	Port role:
00	Unknown
01	Alternate port
10	Root port
11	Designated port
4	Learning
5	Forwarding
6	Agreement
7	Topology change acknowledgement (TCA)

The sending switch sets the proposal flag in the RSTP BPDU to propose itself as the designated switch on that LAN. The port role in the proposal message is always set to the designated port.

The sending switch sets the agreement flag in the RSTP BPDU to accept the previous proposal. The port role in the agreement message is always set to the root port.

The RSTP does not have a separate topology change notification (TCN) BPDU. It uses the topology change (TC) flag to show the topology changes. However, for interoperability with 802.1D switches, the RSTP switch processes and generates TCN BPDUs.

The learning and forwarding flags are set according to the state of the sending port.

Processing Superior BPDU Information

If a port receives superior root information (lower bridge ID, lower path cost, and so forth) than currently stored for the port, the RSTP triggers a reconfiguration. If the port is proposed and is selected as the new root port, RSTP forces all the other ports to synchronize.

If the BPDU received is an RSTP BPDU with the proposal flag set, the switch sends an agreement message after all of the other ports are synchronized. If the BPDU is an 802.1D BPDU, the switch does not set the proposal flag and starts the forward-delay timer for the port. The new root port requires twice the forward-delay time to transition to the forwarding state.

If the superior information received on the port causes the port to become a backup or alternate port, RSTP sets the port to the blocking state but does not send the agreement message. The designated port continues sending BPDUs with the proposal flag set until the forward-delay timer expires, at which time the port transitions to the forwarding state.

Processing Inferior BPDU Information

If a designated port receives an inferior BPDU (higher bridge ID, higher path cost, and so forth than currently stored for the port) with a designated port role, it immediately replies with its own information.

Topology Changes

This section describes the differences between the RSTP and the IEEE 802.1D in handling spanning-tree topology changes.

- Detection—Unlike IEEE 802.1D in which *any* transition between the blocking and the forwarding state causes a topology change, *only* transitions from the blocking to the forwarding state cause a topology change with RSTP (only an increase in connectivity is considered a topology change). State changes on an edge port do not cause a topology change. When an RSTP switch detects a topology change, it flushes the learned information on all of its nonedge ports except on those from which it received the TC notification.
- Notification—Unlike IEEE 802.1D, which uses TCN BPDUs, the RSTP does not use them. However, for 802.1D interoperability, an RSTP switch processes and generates TCN BPDUs.
- Acknowledgement—When an RSTP switch receives a TCN message on a designated port from an IEEE 802.1D switch, it replies with an 802.1D configuration BPDU with the TCA bit set. However, if the TC-while timer (the same as the topology-change timer in 802.1D) is active on a root port connected to an 802.1D switch and a configuration BPDU with the TCA bit set is received, the TC-while timer is reset.

 This behavior is only required to support IEEE 802.1D switches. The RSTP BPDUs never have the TCA bit set.

- Propagation—When an RSTP switch receives a TC message from another switch through a designated or root port, it propagates the change to all of its nonedge, designated ports and to the root port (excluding the port on which it is received). The switch starts the TC-while timer for all such ports and flushes the information learned on them.
- Protocol migration—For backward compatibility with IEEE 802.1D switches, RSTP selectively sends IEEE 802.1D configuration BPDUs and TCN BPDUs on a per-port basis.

 When a port is initialized, the migrate-delay timer is started (specifies the minimum time during which RSTP BPDUs are sent), and RSTP BPDUs are sent. While this timer is active, the switch processes all BPDUs received on that port and ignores the protocol type.

 If the switch receives an IEEE 802.1D BPDU after the port's migration-delay timer has expired, it assumes that it is connected to an IEEE 802.1D switch and starts using only IEEE 802.1D BPDUs. However, if the RSTP switch is using IEEE 802.1D BPDUs on a port and receives an RSTP BPDU after the timer has expired, it restarts the timer and starts using RSTP BPDUs on that port.

Configuring MSTP Features

These sections contain this configuration information:

- Default MSTP Configuration, page 15-12
- MSTP Configuration Guidelines, page 15-12
- Specifying the MST Region Configuration and Enabling MSTP, page 15-13 (required)
- Configuring the Root Switch, page 15-14 (optional)
- Configuring a Secondary Root Switch, page 15-16 (optional)
- Configuring Port Priority, page 15-17 (optional)
- Configuring Path Cost, page 15-18 (optional)
- Configuring the Switch Priority, page 15-19 (optional)
- Configuring the Hello Time, page 15-19 (optional)
- Configuring the Forwarding-Delay Time, page 15-20 (optional)
- Configuring the Maximum-Aging Time, page 15-21 (optional)
- Configuring the Maximum-Hop Count, page 15-21 (optional)
- Specifying the Link Type to Ensure Rapid Transitions, page 15-22 (optional)
- Restarting the Protocol Migration Process, page 15-22 (optional)

Default MSTP Configuration

Table 15-3 shows the default MSTP configuration.

Table 15-3 Default MSTP Configuration

Feature	Default Setting
Spanning-tree mode	Rapid PVST+ (PVST+ and MSTP are disabled).
Switch priority (configurable on a per-CIST port basis)	32768.
Spanning-tree port priority (configurable on a per-CIST port basis)	128.
Spanning-tree port cost (configurable on a per-CIST port basis)	1000 Mbps: 4. 100 Mbps: 19. 10 Mbps: 100.
Hello time	2 seconds.
Forward-delay time	15 seconds.
Maximum-aging time	20 seconds.
Maximum hop count	20 hops.

For information about the supported number of spanning-tree instances, see the "Supported Spanning-Tree Instances" section on page 14-10.

MSTP Configuration Guidelines

These are the configuration guidelines for MSTP:

- MSTP is supported only on NNIs on the switch. UNIs do not participate in MSTP.
- When you enable MST by using the **spanning-tree mode mst** global configuration command, RSTP is automatically enabled.
- For two or more switches to be in the same MST region, they must have the same VLAN-to-instance map, the same configuration revision number, and the same name.
- The switch supports up to 16 MST instances. The number of VLANs that can be mapped to a particular MST instance is unlimited.
- PVST+, rapid PVST+, and MSTP are supported, but only one version can be active at any time. (For example, all VLANs run PVST+, all VLANs run rapid PVST+, or all VLANs run MSTP.) For more information, see the "Spanning-Tree Interoperability and Backward Compatibility" section on page 14-10. For information on the recommended trunk port configuration, see the "Interaction with Other Features" section on page 11-16.
- You can manually configure the MST configuration (region name, revision number, and VLAN-to-instance mapping) on each switch within the MST region by using the command-line interface (CLI) or through the SNMP support.
- For load balancing across redundant paths in the network to work, all VLAN-to-instance mapping assignments must match; otherwise, all traffic flows on a single link.
- All MST boundary ports must be forwarding for load balancing between a PVST+ and an MST cloud or between a rapid-PVST+ and an MST cloud. For this to occur, the IST master of the MST cloud should also be the root of the CST. If the MST cloud consists of multiple MST regions, one

of the MST regions must contain the CST root, and all of the other MST regions must have a better path to the root contained within the MST cloud than a path through the PVST+ or rapid-PVST+ cloud. You might have to manually configure the switches in the clouds.

- Partitioning the network into a large number of regions is not recommended. However, if this situation is unavoidable, we recommend that you partition the switched LAN into smaller LANs interconnected by routers or non-Layer 2 devices.

Specifying the MST Region Configuration and Enabling MSTP

For two or more switches to be in the same MST region, they must have the same VLAN-to-instance mapping, the same configuration revision number, and the same name.

A region can have one member or multiple members with the same MST configuration; each member must be capable of processing RSTP BPDUs. There is no limit to the number of MST regions in a network, but each region can support up to 16 spanning-tree instances. You can assign a VLAN to only one spanning-tree instance at a time.

Beginning in privileged EXEC mode, follow these steps to specify the MST region configuration and enable MSTP. This procedure is required.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree mst configuration**	Enter MST configuration mode.
Step 3	**instance** *instance-id* **vlan** *vlan-range*	Map VLANs to an MST instance. • For *instance-id*, the range is 1 to 15. • For **vlan** *vlan-range*, the range is 1 to 4094. When you map VLANs to an MST instance, the mapping is incremental, and the VLANs specified in the command are added to or removed from the VLANs that were previously mapped. To specify a VLAN range, use a hyphen; for example, **instance 1 vlan 1-63** maps VLANs 1 through 63 to MST instance 1. To specify a VLAN series, use a comma; for example, **instance 1 vlan 10, 20, 30** maps VLANs 10, 20, and 30 to MST instance 1.
Step 4	**name** *name*	Specify the configuration name. The *name* string has a maximum length of 32 characters and is case sensitive.
Step 5	**revision** *version*	Specify the configuration revision number. The range is 0 to 65535.
Step 6	**show pending**	Verify your configuration by displaying the pending configuration.
Step 7	**exit**	Apply all changes, and return to global configuration mode.

	Command	Purpose
Step 8	**spanning-tree mode mst**	Enable MSTP. RSTP is also enabled. **Caution** Changing spanning-tree modes can disrupt traffic because all spanning-tree instances are stopped for the previous mode and restarted in the new mode. You cannot run both MSTP and rapid PVST+ or both MSTP and PVST+ at the same time.
Step 9	**end**	Return to privileged EXEC mode.
Step 10	**show running-config**	Verify your entries.
Step 11	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default MST region configuration, use the **no spanning-tree mst configuration** global configuration command. To return to the default VLAN-to-instance map, use the **no instance** *instance-id* [**vlan** *vlan-range*] MST configuration command. To return to the default name, use the **no name** MST configuration command. To return to the default revision number, use the **no revision** MST configuration command. To re-enable rapid PVST+, use the **no spanning-tree mode** or the **spanning-tree mode pvst** global configuration command.

This example shows how to enter MST configuration mode, map VLANs 10 to 20 to MST instance 1, name the region *region1*, set the configuration revision to 1, display the pending configuration, apply the changes, and return to global configuration mode:

```
Switch(config)# spanning-tree mst configuration
Switch(config-mst)# instance 1 vlan 10-20
Switch(config-mst)# name region1
Switch(config-mst)# revision 1
Switch(config-mst)# show pending
Pending MST configuration
Name      [region1]
Revision  1
Instance  Vlans Mapped
--------  --------------------
0         1-9,21-4094
1         10-20
-------------------------------

Switch(config-mst)# exit
Switch(config)#
```

Configuring the Root Switch

The switch maintains a spanning-tree instance for the group of VLANs mapped to it. A bridge ID, consisting of the switch priority and the switch MAC address, is associated with each instance. For a group of VLANs, the switch with the lowest bridge ID becomes the root switch.

To configure a switch to become the root, use the **spanning-tree mst** *instance-id* **root** global configuration command to modify the switch priority from the default value (32768) to a significantly lower value so that the switch becomes the root switch for the specified spanning-tree instance. When you enter this command, the switch checks the switch priorities of the root switches. Because of the extended system ID support, the switch sets its own priority for the specified instance to 24576 if this value will cause this switch to become the root for the specified spanning-tree instance.

If any root switch for the specified instance has a switch priority lower than 24576, the switch sets its own priority to 4096 less than the lowest switch priority. (4096 is the value of the least-significant bit of a 4-bit switch priority value as shown in Table 14-1 on page 14-4.)

If your network consists of switches that both do and do not support the extended system ID, it is unlikely that the switch with the extended system ID support will become the root switch. The extended system ID increases the switch priority value every time the VLAN number is greater than the priority of the connected switches running older software.

The root switch for each spanning-tree instance should be a backbone or distribution switch. Do not configure an access switch as the spanning-tree primary root.

Use the **diameter** keyword, which is available only for MST instance 0, to specify the Layer 2 network diameter (that is, the maximum number of switch hops between any two end stations in the Layer 2 network). When you specify the network diameter, the switch automatically sets an optimal hello time, forward-delay time, and maximum-age time for a network of that diameter, which can significantly reduce the convergence time. You can use the **hello** keyword to override the automatically calculated hello time.

Note After configuring the switch as the root switch, we recommend that you avoid manually configuring the hello time, forward-delay time, and maximum-age time through the **spanning-tree mst hello-time**, **spanning-tree mst forward-time**, and the **spanning-tree mst max-age** global configuration commands.

Beginning in privileged EXEC mode, follow these steps to configure a switch as the root switch. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree mst** *instance-id* **root primary** [**diameter** *net-diameter* [**hello-time** *seconds*]]	Configure a switch as the root switch. • For *instance-id*, you can specify a single instance, a range of instances separated by a hyphen, or a series of instances separated by a comma. The range is 0 to 15. • (Optional) For **diameter** *net-diameter*, specify the maximum number of switches between any two end stations. The range is 2 to 7. This keyword is available only for MST instance 0. • (Optional) For **hello-time** *seconds*, specify the interval in seconds between the generation of configuration messages by the root switch. The range is 1 to 10 seconds; the default is 2 seconds.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show spanning-tree mst** ***instance-id***	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the switch to its default setting, use the **no spanning-tree mst** *instance-id* **root** global configuration command.

Configuring a Secondary Root Switch

When you configure a switch with the extended system ID support as the secondary root, the switch priority is modified from the default value (32768) to 28672. The switch is then likely to become the root switch for the specified instance if the primary root switch fails. This is assuming that the other network switches use the default switch priority of 32768 and therefore are unlikely to become the root switch.

You can execute this command on more than one switch to configure multiple backup root switches. Use the same network diameter and hello-time values that you used when you configured the primary root switch with the **spanning-tree mst** *instance-id* **root primary** global configuration command.

Beginning in privileged EXEC mode, follow these steps to configure a switch as the secondary root switch. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree mst** *instance-id* **root secondary** [**diameter** *net-diameter* [**hello-time** *seconds*]]	Configure a switch as the secondary root switch. • For *instance-id*, you can specify a single instance, a range of instances separated by a hyphen, or a series of instances separated by a comma. The range is 0 to 15. • (Optional) For **diameter** *net-diameter*, specify the maximum number of switches between any two end stations. The range is 2 to 7. This keyword is available only for MST instance 0. • (Optional) For **hello-time** *seconds*, specify the interval in seconds between the generation of configuration messages by the root switch. The range is 1 to 10 seconds; the default is 2 seconds. Use the same network diameter and hello-time values that you used when configuring the primary root switch. See the "Configuring the Root Switch" section on page 15-14.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show spanning-tree mst** ***instance-id***	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the switch to its default setting, use the **no spanning-tree mst** *instance-id* **root** global configuration command.

Configuring Port Priority

If a loop occurs, the MSTP uses the port priority when selecting an NNI to put into the forwarding state. You can assign higher priority values (lower numerical values) to NNIs that you want selected first and lower priority values (higher numerical values) that you want selected last. If all interfaces have the same priority value, the MSTP puts the interface with the lowest interface number in the forwarding state and blocks the other interfaces.

Beginning in privileged EXEC mode, follow these steps to configure the MSTP port priority of an interface. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify an interface to configure, and enter interface configuration mode. Valid interfaces include physical NNIs and NNI port channels. The port-channel range is 1 to 48. **Note** If a physical interface is not an NNI, enter the **port-type nni** interface configuration command before attempting to configure MST port priority. If the interface is a port channel, all members of the port channel must be NNIs.
Step 3	**spanning-tree mst** *instance-id* **port-priority** *priority*	Configure the port priority. • For *instance-id*, you can specify a single instance, a range of instances separated by a hyphen, or a series of instances separated by a comma. The range is 0 to 15. • For *priority*, the range is 0 to 240 in increments of 16. The default is 128. The lower the number, the higher the priority. The priority values are 0, 16, 32, 48, 64, 80, 96, 112, 128, 144, 160, 176, 192, 208, 224, and 240. All other values are rejected.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show spanning-tree mst interface** *interface-id* or **show spanning-tree mst** *instance-id*	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Note The **show spanning-tree mst interface** *interface-id* privileged EXEC command displays information only if the port is in a link-up operative state. Otherwise, you can use the **show running-config interface** privileged EXEC command to confirm the configuration.

To return the interface to its default setting, use the **no spanning-tree mst** *instance-id* **port-priority** interface configuration command.

Configuring Path Cost

The MSTP path cost default value is derived from the media speed of an NNI. If a loop occurs, the MSTP uses cost when selecting an interface to put in the forwarding state. You can assign lower cost values to NNIs that you want selected first and higher cost values that you want selected last. If all interfaces have the same cost value, the MSTP puts the interface with the lowest interface number in the forwarding state and blocks the other interfaces.

Beginning in privileged EXEC mode, follow these steps to configure the MSTP cost of an interface. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify an interface to configure, and enter interface configuration mode. Valid interfaces include physical NNIs and NNI port-channel logical interfaces. The port-channel range is 1 to 48. **Note** If a physical interface is not an NNI, enter the **port-type nni** interface configuration command before you configure MST path cost. If the interface is a port channel, all members of the port channel must be NNIs.
Step 3	**spanning-tree mst** *instance-id* **cost** *cost*	Configure the cost. If a loop occurs, the MSTP uses the path cost when selecting an interface to place into the forwarding state. A lower path cost represents higher-speed transmission. • For *instance-id*, you can specify a single instance, a range of instances separated by a hyphen, or a series of instances separated by a comma. The range is 0 to 15. • For *cost*, the range is 1 to 200000000; the default value is derived from the media speed of the interface.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show spanning-tree mst interface** *interface-id* or **show spanning-tree mst** *instance-id*	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Note The **show spanning-tree mst interface** *interface-id* privileged EXEC command displays information only for ports that are in a link-up operative state. Otherwise, you can use the **show running-config** privileged EXEC command to confirm the configuration.

To return the interface to its default setting, use the **no spanning-tree mst** *instance-id* **cost** interface configuration command.

Configuring the Switch Priority

You can configure the switch priority and make it more likely that the switch will be chosen as the root switch.

Note Exercise care when using this command. For most situations, we recommend that you use the **spanning-tree mst** *instance-id* **root primary** and the **spanning-tree mst** *instance-id* **root secondary** global configuration commands to modify the switch priority.

Beginning in privileged EXEC mode, follow these steps to configure the switch priority. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree mst** *instance-id* **priority** *priority*	Configure the switch priority. • For *instance-id*, you can specify a single instance, a range of instances separated by a hyphen, or a series of instances separated by a comma. The range is 0 to 15. • For *priority*, the range is 0 to 61440 in increments of 4096; the default is 32768. The lower the number, the more likely the switch will be chosen as the root switch. Priority values are 0, 4096, 8192, 12288, 16384, 20480, 24576, 28672, 32768, 36864, 40960, 45056, 49152, 53248, 57344, and 61440. All other values are rejected.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show spanning-tree mst** *instance-id*	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the switch to its default setting, use the **no spanning-tree mst** *instance-id* **priority** global configuration command.

Configuring the Hello Time

You can configure the interval between the generation of configuration messages by the root switch by changing the hello time.

Note Exercise care when using this command. For most situations, we recommend that you use the **spanning-tree mst** *instance-id* **root primary** and the **spanning-tree mst** *instance-id* **root secondary** global configuration commands to modify the hello time.

Beginning in privileged EXEC mode, follow these steps to configure the hello time for all MST instances. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree mst hello-time** *seconds*	Configure the hello time for all MST instances. The hello time is the interval between the generation of configuration messages by the root switch. These messages mean that the switch is alive. For *seconds*, the range is 1 to 10; the default is 2.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show spanning-tree mst**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the switch to its default setting, use the **no spanning-tree mst hello-time** global configuration command.

Configuring the Forwarding-Delay Time

Beginning in privileged EXEC mode, follow these steps to configure the forwarding-delay time for all MST instances. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree mst forward-time** *seconds*	Configure the forward time for all MST instances. The forward delay is the number of seconds a port waits before changing from its spanning-tree learning and listening states to the forwarding state. For *seconds*, the range is 4 to 30; the default is 15.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show spanning-tree mst**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the switch to its default setting, use the **no spanning-tree mst forward-time** global configuration command.

Configuring the Maximum-Aging Time

Beginning in privileged EXEC mode, follow these steps to configure the maximum-aging time for all MST instances. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree mst max-age** *seconds*	Configure the maximum-aging time for all MST instances. The maximum-aging time is the number of seconds a switch waits without receiving spanning-tree configuration messages before attempting a reconfiguration. For *seconds*, the range is 6 to 40; the default is 20.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show spanning-tree mst**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the switch to its default setting, use the **no spanning-tree mst max-age** global configuration command.

Configuring the Maximum-Hop Count

Beginning in privileged EXEC mode, follow these steps to configure the maximum-hop count for all MST instances. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree mst max-hops** *hop-count*	Specify the number of hops in a region before the BPDU is discarded, and the information held for a port is aged. For *hop-count*, the range is 1 to 40; the default is 20.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show spanning-tree mst**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the switch to its default setting, use the **no spanning-tree mst max-hops** global configuration command.

Specifying the Link Type to Ensure Rapid Transitions

If you connect an NNI to another NNI through a point-to-point link and the local port becomes a designated port, the RSTP negotiates a rapid transition with the other port by using the proposal-agreement handshake to ensure a loop-free topology as described in the "Rapid Convergence" section on page 15-7.

By default, the link type is controlled from the duplex mode of the interface: a full-duplex port is considered to have a point-to-point connection; a half-duplex port is considered to have a shared connection. If you have a half-duplex link physically connected point-to-point to a single port on a remote switch running MSTP, you can override the default setting of the link type and enable rapid transitions to the forwarding state.

Beginning in privileged EXEC mode, follow these steps to override the default link-type setting. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify an interface to configure, and enter interface configuration mode. Valid interfaces include physical NNIs, VLANs, and NNI port-channel logical interfaces. The VLAN ID range is 1 to 4094. The port-channel range is 1 to 48. **Note** If a physical interface is not an NNI, enter the **port-type nni** interface configuration command before you configure the link type. If the interface is a port channel, all members of the port channel must be NNIs.
Step 3	**spanning-tree link-type point-to-point**	Specify that the link type of a port is point-to-point.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show spanning-tree mst interface** *interface-id*	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the port to its default setting, use the **no spanning-tree link-type** interface configuration command.

Restarting the Protocol Migration Process

A switch running MSTP supports a built-in protocol migration mechanism that enables it to interoperate with legacy 802.1D switches. If this switch receives a legacy 802.1D configuration BPDU (a BPDU with the protocol version set to 0), it sends only 802.1D BPDUs on that port. An MSTP switch also can detect that a port is at the boundary of a region when it receives a legacy BPDU, an MST BPDU (Version 3) associated with a different region, or an RST BPDU (Version 2).

However, the switch does not automatically revert to the MSTP mode if it no longer receives 802.1D BPDUs because it cannot detect whether the legacy switch has been removed from the link unless the legacy switch is the designated switch. A switch also might continue to assign a boundary role to a port when the switch to which it is connected has joined the region.

To restart the protocol migration process (force the renegotiation with neighboring switches) on the switch, use the **clear spanning-tree detected-protocols** privileged EXEC command.

To restart the protocol migration process on a specific interface, use the **clear spanning-tree detected-protocols interface** *interface-id* privileged EXEC command.

Displaying the MST Configuration and Status

To display the spanning-tree status, use one or more of the privileged EXEC commands in Table 15-4:

Table 15-4 Commands for Displaying MST Status

Command	Purpose
show spanning-tree mst configuration	Displays the MST region configuration.
show spanning-tree mst *instance-id*	Displays MST information for the specified instance.
show spanning-tree mst interface *interface-id*	Displays MST information for the specified interface.

For information about other keywords for the **show spanning-tree** privileged EXEC command, see the command reference for this release.



CHAPTER 16

Configuring Optional Spanning-Tree Features

This chapter describes how to configure optional spanning-tree features on the Cisco ME 3400 Ethernet Access switch. You can configure all of these features when your switch is running per-VLAN spanning-tree plus (PVST+). You can configure only the noted features when your switch is running the Multiple Spanning Tree Protocol (MSTP) or the rapid per-VLAN spanning-tree plus (rapid-PVST+) protocol. On the Cisco ME switch, STP is supported only on network node interfaces (NNIs). User network interfaces (UNIs) on the switch do not participate in STP and forward traffic immediately when they are brought up.

For information on configuring the PVST+ and rapid PVST+, see Chapter 14, "Configuring STP." For information about the Multiple Spanning Tree Protocol (MSTP) and how to map multiple VLANs to the same spanning-tree instance, see Chapter 15, "Configuring MSTP."

Note For complete syntax and usage information for the commands used in this chapter, see the command reference for this release.

This chapter consists of these sections:

- Understanding Optional Spanning-Tree Features, page 16-1
- Configuring Optional Spanning-Tree Features, page 16-5
- Displaying the Spanning-Tree Status, page 16-10

Understanding Optional Spanning-Tree Features

These sections contain this conceptual information:

- Understanding Port Fast, page 16-2
- Understanding BPDU Guard, page 16-2
- Understanding BPDU Filtering, page 16-3
- Understanding EtherChannel Guard, page 16-3
- Understanding Root Guard, page 16-3
- Understanding Loop Guard, page 16-4

Understanding Port Fast

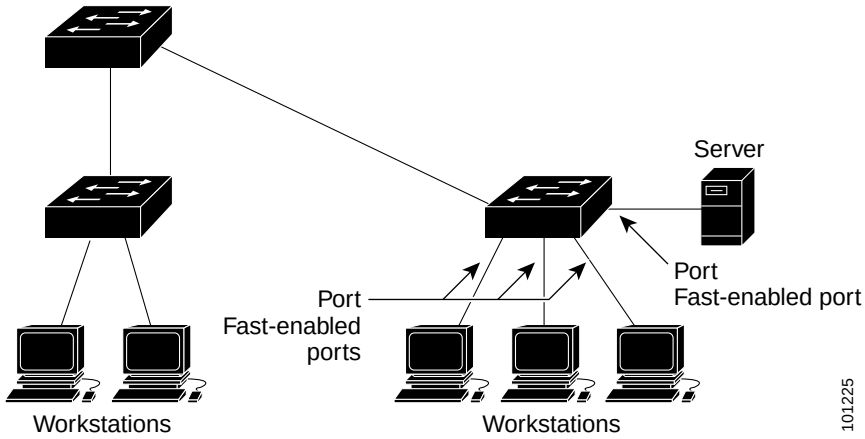
Port Fast immediately brings an NNI configured as an access or trunk port to the forwarding state from a blocking state, bypassing the listening and learning states. You can use Port Fast on NNIs connected to a single workstation or server, as shown in Figure 16-1, to allow those devices to immediately connect to the network, rather than waiting for the spanning tree to converge.

NNIs connected to a single workstation or server should not receive bridge protocol data units (BPDUs). An NNI with Port Fast enabled goes through the normal cycle of spanning-tree status changes when the switch is restarted.

Note Because the purpose of Port Fast is to minimize the time interfaces must wait for spanning tree to converge, it is effective only when used on NNIs connected to end stations. If you enable Port Fast on an interface connecting to another switch, you risk creating a spanning-tree loop.

You can enable this feature by using the **spanning-tree portfast** interface configuration or the **spanning-tree portfast default** global configuration command.

Figure 16-1 Port Fast-Enabled Interfaces



Understanding BPDU Guard

The BPDU guard feature can be globally enabled on the switch or can be enabled per interface, but the feature operates with some differences.

At the global level, you enable BPDU guard on Port Fast-enabled NNIs by using the **spanning-tree portfast bpduguard default** global configuration command. Spanning tree shuts down NNIs that are in a Port Fast-operational state if any BPDU is received on those NNIs. In a valid configuration, Port Fast-enabled NNIs do not receive BPDUs. Receiving a BPDU on a Port Fast-enabled NNI signals an invalid configuration, such as the connection of an unauthorized device, and the BPDU guard feature puts the interface in the error-disabled state.

At the interface level, you enable BPDU guard on any NNI by using the **spanning-tree bpduguard enable** interface configuration command without also enabling the Port Fast feature. When the NNI receives a BPDU, it is put in the error-disabled state.

The BPDU guard feature provides a secure response to invalid configurations because you must manually put the interface back in service. Use the BPDU guard feature in a service-provider network to prevent an access port from participating in the spanning tree.

You can enable the BPDU guard feature for the entire switch or for an interface.

Understanding BPDU Filtering

The BPDU filtering feature can be globally enabled on the switch or can be enabled per interface, but the feature operates with some differences.

At the global level, you can enable BPDU filtering on Port Fast-enabled NNIs by using the **spanning-tree portfast bpdufilter default** global configuration command. This command prevents interfaces that are in a Port Fast-operational state from sending or receiving BPDUs. The interfaces still send a few BPDUs at link-up before the switch begins to filter outbound BPDUs. You should globally enable BPDU filtering on a switch so that hosts connected to these NNIs do not receive BPDUs. If a BPDU is received on a Port Fast-enabled NNI, the interface loses its Port Fast-operational status, and BPDU filtering is disabled.

At the interface level, you can enable BPDU filtering on any NNI by using the **spanning-tree bpdufilter enable** interface configuration command without also enabling the Port Fast feature. This command prevents the interface from sending or receiving BPDUs.

Caution Enabling BPDU filtering on an NNI is the same as disabling spanning tree on it and can result in spanning-tree loops.

You can enable the BPDU filtering feature for the entire switch or for an NNI.

Understanding EtherChannel Guard

You can use EtherChannel guard to detect an EtherChannel misconfiguration between the switch and a connected device. A misconfiguration can occur if the switch NNIs are configured in an EtherChannel, but the interfaces on the other device are not. A misconfiguration can also occur if the channel parameters are not the same at both ends of the EtherChannel. For EtherChannel configuration guidelines, see the "EtherChannel Configuration Guidelines" section on page 31-10.

If the switch detects a misconfiguration on the other device, EtherChannel guard places the switch NNIs in the error-disabled state, and displays an error message.

You can enable this feature by using the **spanning-tree etherchannel guard misconfig** global configuration command.

Understanding Root Guard

The Layer 2 network of a service provider (SP) can include many connections to switches that are not owned by the SP. In such a topology, the spanning tree can reconfigure itself and select a *customer switch* as the root switch, as shown in Figure 16-2. You can avoid this situation by enabling root guard on SP switch interfaces that connect to switches in your customer's network. If spanning-tree calculations cause an interface in the customer network to be selected as the root port, root guard then places the interface in the root-inconsistent (blocked) state to prevent the customer's switch from becoming the root switch or being in the path to the root.

If a switch outside the SP network becomes the root switch, the interface is blocked (root-inconsistent state), and spanning tree selects a new root switch. The customer's switch does not become the root switch and is not in the path to the root.

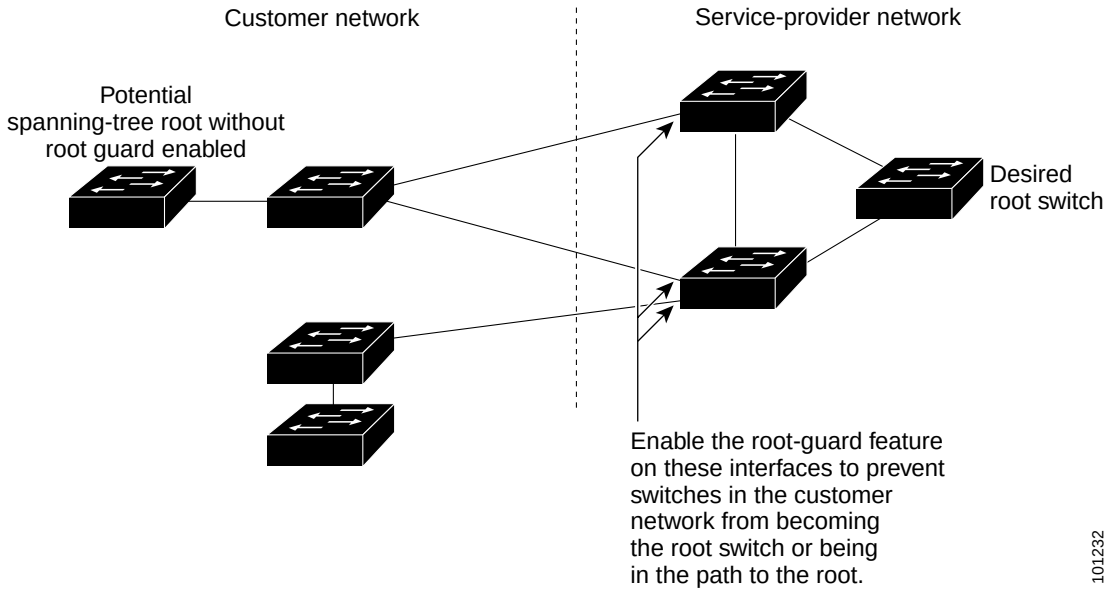
If the switch is operating in multiple spanning-tree (MST) mode, root guard forces the interface to be a designated port. If a boundary port is blocked in an internal spanning-tree (IST) instance because of root guard, the interface also is blocked in all MST instances. A boundary port is an interface that connects to a LAN, the designated switch of which is either an 802.1D switch or a switch with a different MST region configuration.

Root guard enabled on an interface applies to all the VLANs to which the interface belongs. VLANs can be grouped and mapped to an MST instance.

You can enable this feature by using the **spanning-tree guard root** interface configuration command.

Caution Misuse of the root-guard feature can cause a loss of connectivity.

Figure 16-2 Root Guard in a Service-Provider Network



Understanding Loop Guard

You can use loop guard to prevent alternate or root ports from becoming designated ports because of a failure that leads to a unidirectional link. This feature is most effective when it is enabled on the entire switched network. Loop guard prevents alternate and root ports from becoming designated ports, and spanning tree does not send BPDUs on root or alternate ports.

You can enable this feature by using the **spanning-tree loopguard default** global configuration command.

When the switch is operating in PVST+ or rapid-PVST+ mode, loop guard prevents alternate and root ports from becoming designated ports, and spanning tree does not send BPDUs on root or alternate ports.

When the switch is operating in MST mode, BPDUs are not sent on nonboundary ports only if the interface is blocked by loop guard in all MST instances. On a boundary port, loop guard blocks the interface in all MST instances.

Configuring Optional Spanning-Tree Features

These sections contain this configuration information:

- Default Optional Spanning-Tree Configuration, page 16-5
- Optional Spanning-Tree Configuration Guidelines, page 16-5
- Enabling Port Fast, page 16-5 (optional)
- Enabling BPDU Guard, page 16-6 (optional)
- Enabling BPDU Filtering, page 16-7 (optional)
- Enabling EtherChannel Guard, page 16-8 (optional)
- Enabling Root Guard, page 16-9 (optional)
- Enabling Loop Guard, page 16-9 (optional)

Default Optional Spanning-Tree Configuration

Table 16-1 shows the default optional spanning-tree configuration. Only NNIs participate in STP on the switch. UNIs are always in the forwarding state.

Table 16-1 Default Optional Spanning-Tree Configuration

Feature	Default Setting
Port Fast, BPDU filtering, BPDU guard	Globally disabled (unless they are individually configured per NNI).
EtherChannel guard	Globally enabled.
Root guard	Disabled on all NNIs.
Loop guard	Disabled on all NNIs.

Optional Spanning-Tree Configuration Guidelines

You can configure PortFast, BPDU guard, BPDU filtering, EtherChannel guard, root guard, or loop guard if your switch is running PVST+, rapid PVST+, or MSTP.

Optional spanning-tree configuration commands are not supported on UNIs.

Enabling Port Fast

An NNI with the Port Fast feature enabled is moved directly to the spanning-tree forwarding state without waiting for the standard forward-time delay.

Caution Use Port Fast *only* when connecting a single end station to an access or trunk port. Enabling this feature on an interface connected to a switch or hub could prevent spanning tree from detecting and disabling loops in your network, which could cause broadcast storms and address-learning problems.

You can enable this feature if your switch is running PVST+, rapid PVST+, or MSTP.

Beginning in privileged EXEC mode, follow these steps to enable Port Fast. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify an interface to configure, and enter interface configuration mode. If the interface is not an NNI, you must enter the **port-type nni** interface configuration command before enabling Port Fast.
Step 3	**spanning-tree portfast** [**trunk**]	Enable Port Fast on an access port connected to a single workstation or server. By specifying the **trunk** keyword, you can enable Port Fast on a trunk port. **Note** To enable Port Fast on trunk ports, you must use the **spanning-tree portfast trunk** interface configuration command. The **spanning-tree portfast** command does not work on trunk ports. **Caution** Make sure that there are no loops in the network between the trunk port and the workstation or server before you enable Port Fast on a trunk port. By default, Port Fast is disabled on all NNIs. UNIs do not participate in STP.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show spanning-tree interface** *interface-id* **portfast**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Note You can use the **spanning-tree portfast default** global configuration command to globally enable the Port Fast feature on all nontrunking NNIs.

To disable the Port Fast feature, use the **spanning-tree portfast disable** interface configuration command.

Enabling BPDU Guard

When you globally enable BPDU guard on NNIs that are Port Fast-enabled (the interfaces are in a Port Fast-operational state), spanning tree shuts down Port Fast-enabled NNIs that receive BPDUs.

In a valid configuration, Port Fast-enabled interfaces do not receive BPDUs. Receiving a BPDU on a Port Fast-enabled interface signals an invalid configuration, such as the connection of an unauthorized device, and the BPDU guard feature puts the interface in the error-disabled state. The BPDU guard feature provides a secure response to invalid configurations because you must manually put the interface back in service. Use the BPDU guard feature in a service-provider network to prevent an access port from participating in the spanning tree.

Caution Configure Port Fast only on NNIs that connect to end stations; otherwise, an accidental topology loop could cause a data packet loop and disrupt switch and network operation.

You also can use the **spanning-tree bpduguard enable** interface configuration command to enable BPDU guard on any NNI without also enabling the Port Fast feature. When the interface receives a BPDU, it is put in the error-disabled state.

You can enable the BPDU guard feature if your switch is running PVST+, rapid PVST+, or MSTP.

Beginning in privileged EXEC mode, follow these steps to globally enable the BPDU guard feature. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree portfast bpduguard default**	Globally enable BPDU guard. (By default, BPDU guard is disabled.) **Note** Globally enabling BPDU guard enables it only on NNIs; the command has no effect on UNIs.
Step 3	**interface** *interface-id*	Specify the interface connected to an end station, and enter interface configuration mode. If the interface is not an NNI, you must enter the **port-type nni** interface configuration command before enabling Port Fast.
Step 4	**spanning-tree portfast**	Enable the Port Fast feature.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable BPDU guard, use the **no spanning-tree portfast bpduguard default** global configuration command.

You can override the setting of the **no spanning-tree portfast bpduguard default** global configuration command by using the **spanning-tree bpduguard enable** interface configuration command on an NNI.

Enabling BPDU Filtering

When you globally enable BPDU filtering on Port Fast-enabled NNIs, it prevents interfaces that are in a Port Fast-operational state from sending or receiving BPDUs. The interfaces still send a few BPDUs at link-up before the switch begins to filter outbound BPDUs. You should globally enable BPDU filtering on a switch so that hosts connected to these interfaces do not receive BPDUs. If a BPDU is received on a Port Fast-enabled NNI, the interface loses its Port Fast-operational status, and BPDU filtering is disabled.

Caution Configure Port Fast only on NNIs that connect to end stations; otherwise, an accidental topology loop could cause a data packet loop and disrupt switch and network operation.

You can also use the **spanning-tree bpdufilter enable** interface configuration command to enable BPDU filtering on any NNI without also enabling the Port Fast feature. This command prevents the NNI from sending or receiving BPDUs.

Caution Enabling BPDU filtering on an NNI is the same as disabling spanning tree on it and can result in spanning-tree loops.

You can enable the BPDU filtering feature if your switch is running PVST+, rapid PVST+, or MSTP.

Beginning in privileged EXEC mode, follow these steps to globally enable the BPDU filtering feature. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree portfast bpdufilter default**	Globally enable BPDU filtering. (By default, BPDU filtering is disabled.) **Note** Globally enabling BPDU filtering enables it only on NNIs; the command has no effect on UNIs.
Step 3	**interface** *interface-id*	Specify the interface connected to an end station, and enter interface configuration mode. If the interface is not an NNI, you must enter the **port-type nni** interface configuration command before enabling Port Fast.
Step 4	**spanning-tree portfast**	Enable the Port Fast feature.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable BPDU filtering, use the **no spanning-tree portfast bpdufilter default** global configuration command.

You can override the setting of the **no spanning-tree portfast bpdufilter default** global configuration command by using the **spanning-tree bpdufilter enable** interface configuration command on an NNI.

Enabling EtherChannel Guard

You can enable EtherChannel guard to detect an EtherChannel misconfiguration if your switch is running PVST+, rapid PVST+, or MSTP.

Beginning in privileged EXEC mode, follow these steps to enable EtherChannel guard. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree etherchannel guard misconfig**	Enable EtherChannel guard.
Step 3	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 4	**show spanning-tree summary**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable the EtherChannel guard feature, use the **no spanning-tree etherchannel guard misconfig** global configuration command.

You can use the **show interfaces status err-disabled** privileged EXEC command to show which switch NNIs are disabled because of an EtherChannel misconfiguration. On the remote device, you can enter the **show etherchannel summary** privileged EXEC command to verify the EtherChannel configuration.

After the configuration is corrected, enter the **shutdown** and **no shutdown** interface configuration commands on the port-channel interfaces that were misconfigured.

Enabling Root Guard

Root guard enabled on an NNI applies to all the VLANs to which the NNI belongs.

Note You cannot enable both root guard and loop guard at the same time.

You can enable this feature if your switch is running PVST+, rapid PVST+, or MSTP.

Beginning in privileged EXEC mode, follow these steps to enable root guard on an interface. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify an interface to configure, and enter interface configuration mode. If the interface is not an NNI, you must enter the **port-type nni** interface configuration command before enabling root guard.
Step 3	**spanning-tree guard root**	Enable root guard on the NNI. By default, root guard is disabled on all interfaces.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable root guard, use the **no spanning-tree guard** interface configuration command.

Enabling Loop Guard

You can use loop guard to prevent alternate or root ports from becoming designated ports because of a failure that leads to a unidirectional link. This feature is most effective when it is configured on the entire switched network. Loop guard operates only on NNIs that are considered point-to-point by the spanning tree.

Note You cannot enable both loop guard and root guard at the same time.

You can enable this feature if your switch is running PVST+, rapid PVST+, or MSTP.

Beginning in privileged EXEC mode, follow these steps to enable loop guard. This procedure is optional.

	Command	Purpose
Step 1	**show spanning-tree active** or **show spanning-tree mst**	Verify which interfaces are alternate or root ports.
Step 2	**configure terminal**	Enter global configuration mode.
Step 3	**spanning-tree loopguard default**	Enable loop guard. By default, loop guard is disabled.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To globally disable loop guard, use the **no spanning-tree loopguard default** global configuration command. You can override the setting of the **no spanning-tree loopguard default** global configuration command by using the **spanning-tree guard loop** interface configuration command on an NNI.

Displaying the Spanning-Tree Status

To display the spanning-tree status, use one or more of the privileged EXEC commands in Table 16-2:

Table 16-2 Commands for Displaying the Spanning-Tree Status

Command	Purpose
show spanning-tree active	Displays spanning-tree information on active interfaces only.
show spanning-tree detail	Displays a detailed summary of interface information.
show spanning-tree interface *interface-id*	Displays spanning-tree information for the specified interface.
show spanning-tree mst interface *interface-id*	Displays MST information for the specified interface.
show spanning-tree summary [**totals**]	Displays a summary of interface states or displays the total lines of the spanning-tree state section.

You can clear spanning-tree counters by using the **clear spanning-tree** [**interface** *interface-id*] privileged EXEC command.

For information about other keywords for the **show spanning-tree** privileged EXEC command, see the command reference for this release.



Configuring Flex Links

This chapter describes how to configure Flex Links, a pair of interfaces on the Cisco ME 3400 switch that are used to provide a mutual backup. This feature is available only when the switch is running the metro IP access or metro access image.

Note For complete syntax and usage information for the commands used in this chapter, see the command reference for this release.

The chapter consists of these sections:

- Understanding Flex Links, page 17-1
- Configuring Flex Links, page 17-2
- Monitoring Flex Links, page 17-4

Understanding Flex Links

Flex Links are a pair of a Layer 2 interfaces (switchports or port channels), where one interface is configured to act as a backup to the other. The feature provides an alternative solution to the Spanning Tree Protocol (STP), allowing users to turn off STP and still provide basic link redundancy. Flex Links are typically configured in service provider or enterprise networks where customers do not want to run STP on the switch. If the switch is running STP, it is not necessary to configure Flex Links because STP already provides link-level redundancy or backup.

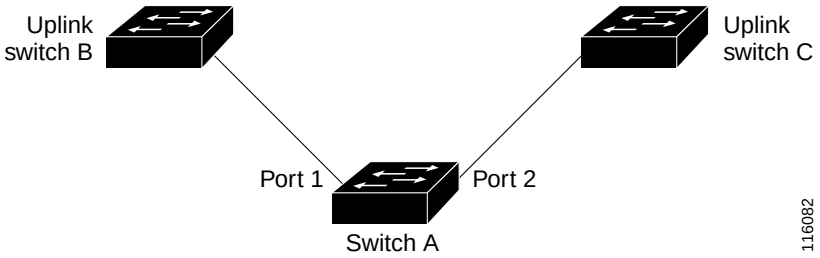
Note STP is available only on network node interfaces (NNIs)

You configure Flex Links on one Layer 2 interface (the active link) by assigning another Layer 2 interface as the Flex Link or backup link. When one of the links is up and forwarding traffic, the other link is in standby mode, ready to begin forwarding traffic if the other link shuts down. At any given time, only one of the interfaces is in the linkup state and forwarding traffic. If the primary link shuts down, the standby link starts forwarding traffic. When the active link comes back up, it goes into standby mode and does not forward traffic. STP is disabled on Flex Link interfaces.

In Figure 17-1, ports 1 and 2 on switch A are connected to uplink switches B and C. Because they are configured as Flex Links, only one of the interfaces is forwarding traffic; the other is in standby mode. If port 1 is the active link, it begins forwarding traffic between port 1 and switch B; the link between

port 2 (the backup link) and switch C is not forwarding traffic. If port 1 goes down, port 2 comes up and starts forwarding traffic to switch C. When port 1 comes back up, it goes into standby mode and does not forward traffic; port 2 continues forwarding traffic.

Figure 17-1 Flex Links Configuration Example



If a primary (forwarding) link goes down, a trap notifies the network management stations. If the standby link goes down, a trap notifies the users.

Flex Links are supported only on Layer 2 ports and port channels, not on VLANs or Layer 3 ports.

Configuring Flex Links

These sections contain this configuration information:

- Default Flex Link Configuration, page 17-2
- Flex Link Configuration Guidelines, page 17-2
- Configuring Flex Links, page 17-3

Default Flex Link Configuration

By default, Flex Links are not configured, and there are no backup interfaces defined.

Flex Link Configuration Guidelines

Follow these guidelines to configure Flex Links:

- You can configure only one Flex Link backup link for any active link, and it must be a different interface from the active interface.
- An interface can belong to only one Flex Link pair. An interface can be a backup link for only one active link. An active link cannot belong to another Flex Link pair.
- Neither of the links can be a port that belongs to an EtherChannel. However, you can configure two port channels (EtherChannel logical interfaces) as Flex Links, and you can configure a port channel and a physical interface as Flex Links, with either the port channel or the physical interface as the active link.

- A backup link does not have to be the same type (Fast Ethernet, Gigabit Ethernet, or port channel) as the active link. However, you should configure both Flex Links with similar characteristics so that there are no loops or changes in behavior if the standby link begins to forward traffic.
- STP is disabled on Flex Link ports. If STP is configured on the switch, Flex Links do not participate in STP in all VLANs in which STP is configured. With STP not running, be sure that there are no loops in the configured topology.\

Note STP is available only on NNIs.

Configuring Flex Links

Beginning in privileged EXEC mode, follow these steps to configure a pair of Flex Links:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface, and enter interface configuration mode. The interface can be a physical Layer 2 interface or a port channel (logical interface). The port-channel range is 1 to 48.
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled, and NNIs are enabled.
Step 4	**switchport backup interface** *interface-id*	Configure a physical Layer 2 interface (or port channel) as part of a Flex Link pair with the interface. When one link is forwarding traffic, the other interface is in standby mode.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show interface** [*interface-id*] **switchport backup**	Verify the configuration.
Step 7	**copy running-config startup config**	(Optional) Save your entries in the switch startup configuration file.

This example shows how to configure an interface with a backup interface and to verify the configuration:

```
Switch# configure terminal
Switch(conf)# interface fastethernet0/1
Switch(conf-if)# no shutdown
Switch(conf-if)# switchport backup interface fastethernet0/2
Switch(conf-if)# end
Switch# show interface switchport backup
Switch Backup Interface Pairs:

Active Interface        Backup Interface        State
------------------------------------------------------------------------------------------
FastEthernet0/1         FastEthernet0/2         Active Up/Backup Standby
FastEthernet0/3         FastEthernet0/4         Active Up/Backup Standby
Port-channel1           GigabitEthernet0/1      Active Up/Backup Standby
```

Monitoring Flex Links

Table 17-1 shows the privileged EXEC command for monitoring Flex Link configuration.

Table 17-1 Flex Link Monitoring Command

Command	Purpose
show interface [*interface-id*] **switchport backup**	Displays the Flex Link backup interface configured for an interface, or displays all Flex Links configured on the switch and the state of each active and backup interface (up or standby mode).



CHAPTER 18

Configuring DHCP Features and IP Source Guard

This chapter describes how to configure DHCP snooping and the option-82 data insertion features on the Cisco ME 3400 Ethernet Access witch. It also describes how to configure the IP source guard feature, which is supported on switches running the metro access and metro IP access images.

Note For complete syntax and usage information for the commands used in this chapter, see the command reference for this release, and see the "DHCP Commands" section in the *Cisco IOS IP Command Reference, Volume 1 of 3: Addressing and Services, Release 12.2*.

This chapter consists of these sections:

- Understanding DHCP Features, page 18-1
- Configuring DHCP Features, page 18-6
- Displaying DHCP Snooping Information, page 18-13
- Understanding IP Source Guard, page 18-13
- Configuring IP Source Guard, page 18-14
- Displaying IP Source Guard Information, page 18-16

Understanding DHCP Features

DHCP is widely used in LAN environments to dynamically assign host IP addresses from a centralized server, which significantly reduces the overhead of administration of IP addresses. DHCP also helps conserve the limited IP address space because IP addresses no longer need to be permanently assigned to hosts; only those hosts that are connected to the network consume IP addresses.

These sections contain this information:

- DHCP Server, page 18-2
- DHCP Relay Agent, page 18-2
- DHCP Snooping, page 18-2
- Option-82 Data Insertion, page 18-3
- DHCP Snooping Binding Database, page 18-5

For information about the DHCP client, see the "*Configuring DHCP*" section of the "*IP Addressing and Services*" section of the *Cisco IOS IP Configuration Guide, Release 12.2*.

DHCP Server

The DHCP server assigns IP addresses from specified address pools on a switch or router to DHCP clients and manages them. If the DHCP server cannot give the DHCP client the requested configuration parameters from its database, it can forward the request to one or more secondary DHCP servers defined by the network administrator. The Cisco ME switch cannot be a DHCP server.

DHCP Relay Agent

A DHCP relay agent is a Layer 3 device that forwards DHCP packets between clients and servers. Relay agents forward requests and replies between clients and servers when they are not on the same physical subnet. Relay agent forwarding is different from the normal Layer 2 forwarding, in which IP datagrams are switched transparently between networks. Relay agents receive DHCP messages and generate new DHCP messages to send on egress interfaces.

DHCP Snooping

DHCP snooping is a DHCP security feature that provides network security by filtering untrusted DHCP messages and by building and maintaining a DHCP snooping binding database, also referred to as a DHCP snooping binding table. For more information about this database, see the "Displaying DHCP Snooping Information" section on page 18-13.

DHCP snooping acts like a firewall between untrusted hosts and DHCP servers. You use DHCP snooping to differentiate between untrusted interfaces connected to the end user and trusted interfaces connected to the DHCP server or another switch.

Note For DHCP snooping to function properly, all DHCP servers must be connected to the switch through trusted interfaces.

An untrusted DHCP message is a message that is received from outside the network or firewall. When you use DHCP snooping in a service-provider environment, an untrusted message is sent from a device that is not in the service-provider network, such as a customer's switch. Messages from unknown devices are untrusted because they can be sources of traffic attacks.

The DHCP snooping binding database has the MAC address, the IP address, the lease time, the binding type, the VLAN number, and the interface information that corresponds to the local untrusted interfaces of a switch. It does not have information regarding hosts interconnected with a trusted interface.

In a service-provider network, a trusted interface is connected to a port on a device in the same network. An untrusted interface is connected to an untrusted interface in the network or to an interface on a device that is not in the network.

When a switch receives a packet on an untrusted interface and the interface belongs to a VLAN in which DHCP snooping is enabled, the switch compares the source MAC address and the DHCP client hardware address. If the addresses match (the default), the switch forwards the packet. If the addresses do not match, the switch drops the packet.

The switch drops a DHCP packet when one of these situations occurs:

- A packet from a DHCP server, such as a DHCPOFFER, DHCPACK, DHCPNAK, or DHCPLEASEQUERY packet, is received from outside the network or firewall.
- A packet is received on an untrusted interface, and the source MAC address and the DHCP client hardware address do not match.
- The switch receives a DHCPRELEASE or DHCPDECLINE broadcast message that has a MAC address in the DHCP snooping binding database, but the interface information in the binding database does not match the interface on which the message was received.
- A DHCP relay agent forwards a DHCP packet that includes a relay-agent IP address that is not 0.0.0.0, or the relay agent forwards a packet that includes option-82 information to an untrusted port.

If the switch is an aggregation switch supporting DHCP snooping and is connected to an edge switch that is inserting DHCP option-82 information, the switch drops packets with option-82 information when packets are received on an untrusted interface. If DHCP snooping is enabled and packets are received on a trusted port, the aggregation switch does not learn the DHCP snooping bindings for connected devices and cannot build a complete DHCP snooping binding database.

When an aggregation switch can be connected to an edge switch through an untrusted interface and you enter the **ip dhcp snooping information option allowed-trust** global configuration command, the aggregation switch accepts packets with option-82 information from the edge switch. The aggregation switch learns the bindings for hosts connected through an untrusted switch interface. The DHCP security features, such as dynamic ARP inspection or IP source guard on a Cisco ME 3400 switch running the metro access or metro IP access image, can still be enabled on the aggregation switch while the switch receives packets with option-82 information on ingress untrusted interfaces to which hosts are connected. The port on the edge switch that connects to the aggregation switch must be configured as a trusted interface.

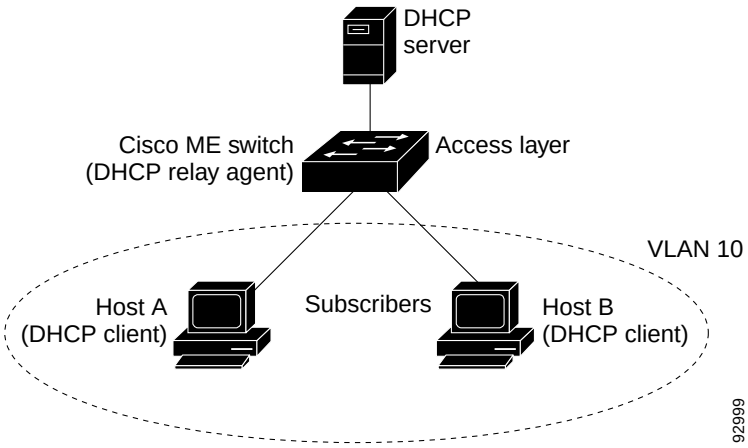
Option-82 Data Insertion

In residential, metropolitan Ethernet-access environments, DHCP can centrally manage the IP address assignments for a large number of subscribers. When the DHCP option-82 feature is enabled on the switch, a subscriber device is identified by the switch port through which it connects to the network (in addition to its MAC address). Multiple hosts on the subscriber LAN can be connected to the same port on the access switch and are uniquely identified.

Note The DHCP option-82 feature is supported only when DHCP snooping is globally enabled and on the VLANs to which subscriber devices using this feature are assigned.

Figure 18-1 is an example of a metropolitan Ethernet network in which a centralized DHCP server assigns IP addresses to subscribers connected to the switch at the access layer. Because the DHCP clients and their associated DHCP server do not reside on the same IP network or subnet, a DHCP relay agent (the Cisco ME switch) is configured with a helper address to enable broadcast forwarding and to transfer DHCP messages between the clients and the server.

Figure 18-1 DHCP Relay Agent in a Metropolitan Ethernet Network



When you enable the DHCP snooping information option 82 on the switch, this sequence of events occurs:

- The host (DHCP client) generates a DHCP request and broadcasts it on the network.
- When the switch receives the DHCP request, it adds the option-82 information in the packet. The option-82 information is the switch MAC address (the remote ID suboption) and the port identifier, **vlan-mod-port**, from which the packet is received (the circuit ID suboption).
- If the IP address of the relay agent is configured, the switch adds this IP address in the DHCP packet.
- The switch forwards the DHCP request that includes the option-82 field to the DHCP server.
- The DHCP server receives the packet. If the server is option-82-capable, it can use the remote ID, the circuit ID, or both to assign IP addresses and implement policies, such as restricting the number of IP addresses that can be assigned to a single remote ID or circuit ID. Then the DHCP server echoes the option-82 field in the DHCP reply.
- The DHCP server unicasts the reply to the switch if the request was relayed to the server by the switch. The switch verifies that it originally inserted the option-82 data by inspecting the remote ID and possibly the circuit ID fields. The switch removes the option-82 field and forwards the packet to the switch port that connects to the DHCP client that sent the DHCP request.

When the previously described sequence of events occurs, the values in these fields in Figure 18-2 do not change:

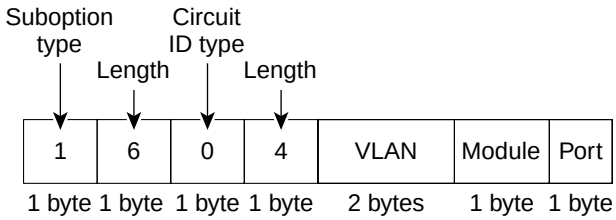
- Circuit ID suboption fields
 - Suboption type
 - Length of the suboption type
 - Circuit ID type
 - Length of the circuit ID type
- Remote ID suboption fields
 - Suboption type
 - Length of the suboption type
 - Remote ID type
 - Length of the circuit ID type

In the port field of the circuit ID suboption, the port numbers start at 3. For example, on a switch with 24 10/100 ports and small form-factor pluggable (SFP) module slots, port 3 is the Fast Ethernet 0/1 port, port 4 is the Fast Ethernet 0/2 port, and so forth. Port 27 is the SFP module slot 0/1, and so forth.

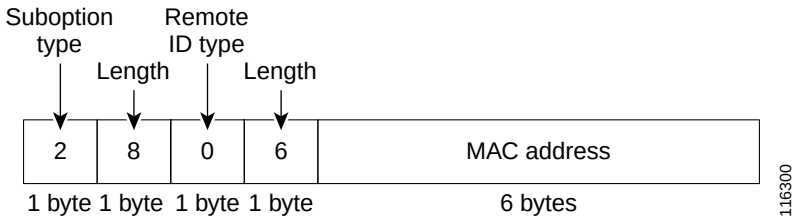
Figure 18-2 shows the packet formats for the remote ID suboption and the circuit ID suboption. The switch uses the packet formats when DHCP snooping is globally enabled and when the **ip dhcp snooping information option** global configuration command is entered.

Figure 18-2 Suboption Packet Formats

Circuit ID Suboption Frame Format



Remote ID Suboption Frame Format



DHCP Snooping Binding Database

When DHCP snooping is enabled, the switch uses the DHCP snooping binding database to store information about untrusted interfaces. The database can have up to 8192 bindings.

Each database entry (*binding*) has an IP address, an associated MAC address, the lease time (in hexadecimal format), the interface to which the binding applies, and the VLAN to which the interface belongs. At the end of each entry is a *checksum* value that accounts for all the bytes associated with the entry. Each entry is 72 bytes, followed by a space and then the checksum value.

To keep the bindings when the switch reloads, you must use the DHCP snooping database agent. If the agent is disabled, dynamic ARP inspection or IP source guard is enabled, and the DHCP snooping binding database has dynamic bindings, the switch loses its connectivity. If the agent is disabled and only DHCP snooping is enabled, the switch does not lose its connectivity, but DHCP snooping might not prevent DHCPP spoofing attacks.

The database agent stores the bindings in a file at a configured location. When reloading, the switch reads the binding file to build the DHCP snooping binding database. The switch keeps the file current by updating it when the database changes.

When a switch learns of new bindings or when it loses bindings, the switch immediately updates the entries in the database. The switch also updates the entries in the binding file. The frequency at which the file is updated is based on a configurable delay, and the updates are batched. If the file is not updated in a specified time (set by the write-delay and abort-timeout values), the update stops.

This is the format of the file that has the bindings:

```
<initial-checksum>
TYPE DHCP-SNOOPING
VERSION 1
BEGIN
<entry-1> <checksum-1>
<entry-2> <checksum-1-2>
...
...
<entry-n> <checksum-1-2-..-n>
END
```

Each entry in the file is tagged with a checksum value that the switch uses to verify the entries when it reads the file. The *initial-checksum* entry on the first line distinguishes entries associated with the latest file update from entries associated with a previous file update.

This is an example of a binding file:

```
2bb4c2a1
TYPE DHCP-SNOOPING
VERSION 1
BEGIN
192.1.168.1 3 0003.47d8.c91f 2BB6488E Fa1/0/4 21ae5fbb
192.1.168.3 3 0003.44d6.c52f 2BB648EB Fa1/0/4 1bdb223f
192.1.168.2 3 0003.47d9.c8f1 2BB648AB Fa1/0/4 584a38f0
END
```

When the switch starts and the calculated checksum value equals the stored checksum value, the switch reads entries from the binding file and adds the bindings to its DHCP snooping binding database. The switch ignores an entry when one of these situations occurs:

- The switch reads the entry and the calculated checksum value does not equal the stored checksum value. The entry and the ones following it are ignored.
- An entry has an expired lease time (the switch might not remove a binding entry when the lease time expires).
- The interface in the entry no longer exists on the system.
- The interface is a routed interface or a DHCP snooping-trusted interface.

Configuring DHCP Features

These sections contain this configuration information:

- Default DHCP Configuration, page 18-7
- DHCP Snooping Configuration Guidelines, page 18-7
- Configuring the DHCP Relay Agent, page 18-8
- Specifying the Packet Forwarding Address, page 18-9
- Enabling DHCP Snooping and Option 82, page 18-10
- Enabling DHCP Snooping on Private VLANs, page 18-11
- Enabling the DHCP Snooping Binding Database Agent, page 18-12

Default DHCP Configuration

Table 18-1 shows the default DHCP configuration.

Table 18-1 Default DHCP Configuration

Feature	Default Setting
DHCP relay agent	Enabled[1]
DHCP packet forwarding address	None configured
Checking the relay agent information	Enabled (invalid messages are dropped)[1]
DHCP relay agent forwarding policy	Replace the existing relay agent information[1]
DHCP snooping enabled globally	Disabled
DHCP snooping information option	Enabled
DHCP snooping option to accept packets on untrusted ingress interfaces[2]	Disabled
DHCP snooping limit rate	None configured
DHCP snooping trust	Untrusted
DHCP snooping VLAN	Disabled
DHCP snooping MAC address verification	Enabled
DHCP snooping binding database agent	Enabled

1. The switch relays DHCP packets only if the IP address of the DHCP server is configured on the SVI of the DHCP client.
2. Use this feature when the switch is an aggregation switch that receives packets with option-82 information from an edge switch.

DHCP Snooping Configuration Guidelines

These are the configuration guidelines for DHCP snooping.

- You must globally enable DHCP snooping on the switch.
- DHCP snooping is not active until DHCP snooping is enabled on a VLAN.
- Before globally enabling DHCP snooping on the switch, make sure that the devices acting as the DHCP server and the DHCP relay agent are configured and enabled.
- When you globally enable DHCP snooping on the switch, these Cisco IOS commands are not available until snooping is disabled. If you enter these commands, the switch returns an error message, and the configuration is not applied.
 - **ip dhcp relay information check** global configuration command
 - **ip dhcp relay information policy** global configuration command
 - **ip dhcp relay information trust-all** global configuration command
 - **ip dhcp relay information trusted** interface configuration command

- Before configuring the DHCP snooping information option on your switch, be sure to configure the device that is acting as the DHCP server. For example, you must specify the IP addresses that the DHCP server can assign or exclude, or you must configure DHCP options for these devices.
- Before configuring the DHCP relay agent on your switch, make sure to configure the device that is acting as the DHCP server. For example, you must specify the IP addresses that the DHCP server can assign or exclude, configure DHCP options for devices, or set up the DHCP database agent.
- If the DHCP relay agent is enabled but DHCP snooping is disabled, the DHCP option-82 data insertion feature is not supported.
- If a switch port is connected to a DHCP server, configure a port as trusted by entering the **ip dhcp snooping trust** interface configuration command.
- If a switch port is connected to a DHCP client, configure a port as untrusted by entering the **no ip dhcp snooping trust** interface configuration command.
- IFollow these guidelines when configuring the DHCP snooping binding database:
 - Because both NVRAM and the flash memory have limited storage capacity, we recommend that you store the binding file on a TFTP server.
 - You must create an empty file at the configured URL on network-based URLs (such as TFTP and FTP) before the switch can initially write bindings to the binding file at that URL for the first time.
 - To ensure that the lease time in the database is accurate, we recommend that NTP is enabled and configured. For more information, see the "Configuring NTP" section on page 5-4.
 - If NTP is configured, the switch writes binding changes to the binding file only when the switch system clock is synchronized with NTP.
- Do not enter the **ip dhcp snooping information option allowed-untrusted** command on an aggregation switch to which an untrusted device is connected. If you enter this command, an untrusted device might spoof the option-82 information.

Configuring the DHCP Relay Agent

Beginning in privileged EXEC mode, follow these steps to enable the DHCP relay agent on the switch:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**service dhcp**	Enable the DHCP relay agent on your switch. By default, this feature is enabled.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable the DHCP relay agent, use the **no service dhcp** global configuration command.

See the "*Configuring DHCP*" section of the "IP Addressing and Services" section of the *Cisco IOS IP Configuration Guide, Release 12.2* for these procedures:

- Checking (validating) the relay agent information
- Configuring the relay agent forwarding policy

Specifying the Packet Forwarding Address

If the DHCP server and the DHCP clients are on different networks or subnets and the switch is running the metro IP access image, you must configure the switch with the **ip helper-address** *address* interface configuration command. The general rule is to configure the command on the Layer 3 interface closest to the client. The address used in the **ip helper-address** command can be a specific DHCP server IP address, or it can be the network address if other DHCP servers are on the destination network segment. Using the network address enables any DHCP server to respond to requests.

Beginning in privileged EXEC mode, follow these steps to specify the packet forwarding address:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface vlan** *vlan-id*	Create a switch virtual interface by entering a VLAN ID, and enter interface configuration mode.
Step 3	**ip address** *ip-address subnet-mask*	Configure the interface with an IP address and an IP subnet.
Step 4	**ip helper-address** *address*	Specify the DHCP packet forwarding address. The helper address can be a specific DHCP server address, or it can be the network address if other DHCP servers are on the destination network segment. Using the network address enables other servers to respond to DHCP requests. If you have multiple servers, you can configure one helper address for each server.
Step 5	**exit**	Return to global configuration mode.
Step 6	**interface range** *port-range* or **interface** *interface-id*	Configure multiple physical ports that are connected to the DHCP clients, and enter interface range configuration mode. or Configure a single physical port that is connected to the DHCP client, and enter interface configuration mode.
Step 7	**no shutdown**	Enable the interface(s), if necessary. By default, user network interfaces (UNIs) are disabled and network node interfaces (NNIs) are enabled.
Step 8	**switchport mode access**	Define the VLAN membership mode for the port.
Step 9	**switchport access vlan** *vlan-id*	Assign the ports to the same VLAN as configured in Step 2.
Step 10	**end**	Return to privileged EXEC mode.
Step 11	**show running-config**	Verify your entries.
Step 12	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the DHCP packet forwarding address, use the **no ip helper-address** *address* interface configuration command.

Enabling DHCP Snooping and Option 82

Beginning in privileged EXEC mode, follow these steps to enable DHCP snooping on the switch:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip dhcp snooping**	Enable DHCP snooping globally.
Step 3	**ip dhcp snooping vlan** *vlan-range*	Enable DHCP snooping on a VLAN or range of VLANs. The range is 1 to 4094. You can enter a single VLAN ID identified by VLAN ID number, a series of VLAN IDs separated by commas, a range of VLAN IDs separated by hyphens, or a range of VLAN IDs separated by entering the starting and ending VLAN IDs separated by a space.
Step 4	**ip dhcp snooping information option**	Enable the switch to insert and remove DHCP relay information (option-82 field) in forwarded DHCP request messages to the DHCP server. The default is enabled.
Step 5	**ip dhcp snooping information option allowed-untrusted**	(Optional) If the switch is an aggregation switch connected to an edge switch, enable the switch to accept incoming DHCP snooping packets with option-82 information from the edge switch. The default is disabled. **Note** You must enter this command only on aggregation switches that are connected to trusted devices.
Step 6	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
Step 7	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled and NNIs are enabled.
Step 8	**ip dhcp snooping trust**	(Optional) Configure the interface as trusted or untrusted. You can use the **no** keyword to configure an interface to receive messages from an untrusted client. The default is untrusted.
Step 9	**ip dhcp snooping limit rate** *rate*	(Optional) Configure the number of DHCP packets per second that an interface can receive. The range is 1 to 2048. By default, no rate limit is configured. **Note** We recommend an untrusted rate limit of not more than 100 packets per second. If you configure rate limiting for trusted interfaces, you might need to increase the rate limit if the port is a trunk port assigned to more than one VLAN on which DHCP snooping is enabled.
Step 10	**exit**	Return to global configuration mode.
Step 11	**ip dhcp snooping verify mac-address**	(Optional) Configure the switch to verify that the source MAC address in a DHCP packet that is received on untrusted ports matches the client hardware address in the packet. The default is to verify that the source MAC address matches the client hardware address in the packet.
Step 12	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 13	**show running-config**	Verify your entries.
Step 14	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable DHCP snooping, use the **no ip dhcp snooping** global configuration command. To disable DHCP snooping on a VLAN or range of VLANs, use the **no ip dhcp snooping vlan** *vlan-range* global configuration command. To disable the insertion and removal of the option-82 field, use the **no ip dhcp snooping information option** global configuration command. To configure an aggregation switch to drop incoming DHCP snooping packets with option-82 information from an edge switch, use the **no ip dhcp snooping information option allowed-untrusted** global configuration command.

This example shows how to enable DHCP snooping globally and on VLAN 10 and to configure a rate limit of 100 packets per second on a port:

```
Switch(config)# ip dhcp snooping
Switch(config)# ip dhcp snooping vlan 10
Switch(config)# ip dhcp snooping information option
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ip dhcp snooping limit rate 100
```

Enabling DHCP Snooping on Private VLANs

You can enable DHCP snooping on private VLANs. If DHCP snooping is enabled, the configuration is propagated to both a primary VLAN and its associated secondary VLANs. If DHCP snooping is enabled on the primary VLAN, it is also configured on the secondary VLANs.

If DHCP snooping is already configured on the primary VLAN and you configure DHCP snooping with different settings on a secondary VLAN, the configuration for the secondary VLAN does not take effect. You must configure DHCP snooping on the primary VLAN. If DHCP snooping is not configured on the primary VLAN, this message appears when you are configuring DHCP snooping on the secondary VLAN, such as VLAN 200:

```
2w5d:%DHCP_SNOOPING-4-DHCP_SNOOPING_PVLAN_WARNING:DHCP Snooping configuration may not take
effect on secondary vlan 200. DHCP Snooping configuration on secondary vlan is derived
from its primary vlan.
```

The **show ip dhcp snooping** privileged EXEC command output shows all VLANs, including primary and secondary private VLANs, on which DHCP snooping is enabled.

Enabling the DHCP Snooping Binding Database Agent

Beginning in privileged EXEC mode, follow these steps to enable and configure the DHCP snooping binding database agent on the switch:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip dhcp snooping database** {**flash:**/*filename* \| **ftp:**//*user*:*password*@*host*/*filename* \| **http:**//[[*username*:*password*]@]{*hostname* \| *host-ip*}[/*directory*] /*image-name*.**tar** \| **rcp:**//*user*@*host*/*filename*}\| **tftp:**//*host*/*filename*	Specify the URL for the database agent or the binding file by using one of these forms: • **flash:**/*filename* • **ftp:**//*user*:*password*@*host*/*filename* • **http:**//[[*username*:*password*]@]{*hostname* \| *host-ip*}[/*directory*] /*image-name*.**tar** • **rcp:**//*user*@*host*/*filename* • **tftp:**//*host*/*filename*
Step 3	**ip dhcp snooping database timeout** *seconds*	Specify when to stop the database transfer process after the binding database changes. The range is from 0 to 86400. Use 0 for an infinite duration. The default is 300 seconds (5 minutes).
Step 4	**ip dhcp snooping database write-delay** *seconds*	Specify the duration for which the transfer should be delayed after the binding database changes. The range is from 15 to 86400 seconds. The default is 300 seconds (5 minutes).
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**ip dhcp snooping binding** *mac-address* **vlan** *vlan-id ip-address* **interface** *interface-id* **expiry** *seconds*	(Optional) Add binding entries to the DHCP snooping binding database. The *vlan-id* range is from 1 to 4904. The *seconds* range is from 1 to 4294967295. Enter this command for each entry that you add. **Note** Use this command when you are testing or debugging the switch.
Step 7	**show ip dhcp snooping database** [**detail**]	Display the status and statistics of the DHCP snooping binding database agent.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To stop using the database agent and binding files, use the **no ip dhcp snooping database** global configuration command.To reset the timeout or delay values, use the **ip dhcp snooping database timeout** *seconds* or the **ip dhcp snooping database write-delay** *seconds* global configuration command.

To clear the statistics of the DHCP snooping binding database agent, use the **clear ip dhcp snooping database statistics** privileged EXEC command. To renew the database, use the **renew ip dhcp snooping database** privileged EXEC command.

To delete binding entries from the DHCP snooping binding database, use the **no ip dhcp snooping binding** *mac-address* **vlan** *vlan-id ip-address* **interface** *interface-id* privileged EXEC command. Enter this command for each entry that you delete.

Displaying DHCP Snooping Information

To display the DHCP snooping information, use one or more of the privileged EXEC commands in Table 18-2:

Table 18-2 Commands for Displaying DHCP Information

Command	Purpose
show ip dhcp snooping	Displays the DHCP snooping configuration for a switch
show ip dhcp snooping binding	Displays only the dynamically configured bindings in the DHCP snooping binding database, also referred to as a binding table.[1]
show ip dhcp snooping database	Displays the DHCP snooping binding database status and statistics.
show ip source binding	Display the dynamically and statically configured bindings.

1. If DHCP snooping is enabled and an interface changes to the down state, the switch does not delete the manually configured bindings.

Understanding IP Source Guard

Note IP source guard is supported only when the metro access or metro IP access image is running on the switch.

IP source guard is a security feature that restricts IP traffic on nonrouted, Layer 2 interfaces by filtering traffic based on the DHCP snooping binding database and on manually configured IP source bindings. You can use IP source guard to prevent traffic attacks caused when a host tries to use the IP address of its neighbor.

You can enable IP source guard when DHCP snooping is enabled on an untrusted interface. After IP source guard is enabled on an interface, the switch blocks all IP traffic received on the interface, except for DHCP packets allowed by DHCP snooping. A port access control list (ACL) is applied to the interface. The port ACL allows only IP traffic with a source IP address in the IP source binding table and denies all other traffic.

The IP source binding table has bindings that are learned by DHCP snooping or are manually configured (static IP source bindings). An entry in this table has an IP address, its associated MAC address, and its associated VLAN number. The switch uses the IP source binding table only when IP source guard is enabled.

IP source guard is supported only on Layer 2 ports, including access and trunk ports.You can configure IP source guard with source IP address filtering or with source IP and MAC address filtering.

These sections contain this information:

- Source IP Address Filtering, page 18-14
- Source IP and MAC Address Filtering, page 18-14

Source IP Address Filtering

When IP source guard is enabled with this option, IP traffic is filtered based on the source IP address. The switch forwards IP traffic when the source IP address matches an entry in the DHCP snooping binding database or a binding in the IP source binding table.

When a DHCP snooping binding or static IP source binding is added, changed, or deleted on an interface, the switch modifies the port ACL using the IP source binding changes, and re-applies the port ACL to the interface.

If you enable IP source guard on an interface on which IP source bindings (dynamically learned by DHCP snooping or manually configured) are not configured, the switch creates and applies a port ACL that denies all IP traffic on the interface. If you disable IP source guard, the switch removes the port ACL from the interface.

Source IP and MAC Address Filtering

When IP source guard is enabled with this option, IP traffic is filtered based on the source IP and MAC addresses. The switch forwards traffic only when the source IP and MAC addresses match an entry in the IP source binding table.

When IP source guard with source IP and MAC address filtering is enabled, the switch filters IP and non-IP traffic. If the source MAC address of an IP or non-IP packet matches a valid IP source binding, the switch forwards the packet. The switch drops all other types of packets except DHCP packets.

The switch uses port security to filter source MAC addresses. The interface can shut down when a port-security violation occurs.

Configuring IP Source Guard

These sections contain this configuration information:

- Default IP Source Guard Configuration, page 18-14
- IP Source Guard Configuration Guidelines, page 18-15
- Enabling IP Source Guard, page 18-15

Default IP Source Guard Configuration

By default, IP source guard is disabled.

IP Source Guard Configuration Guidelines

Note IP source guard is supported only when the metro access or metro IP access image is running on the switch.

These are the configuration guides for IP source guard:

- You can configure static IP bindings only on nonrouted ports. If you enter the **ip source binding** *mac-address* **vlan** *vlan-id ip-address* **interface** *interface-id* global configuration command on a routed interface, this error message appears:

  ```
  Static IP source binding can only be configured on switch port.
  ```

- When IP source guard with source IP filtering is enabled on a VLAN, DHCP snooping must be enabled on the access VLAN to which the interface belongs.
- If you are enabling IP source guard on a trunk interface with multiple VLANs and DHCP snooping is enabled on all the VLANs, the source IP address filter is applied on all the VLANs.

 Note If IP source guard is enabled and you enable or disable DHCP snooping on a VLAN on the trunk interface, the switch might not properly filter traffic.

- When IP source guard with source IP and MAC address filtering is enabled, DHCP snooping and port security must be enabled on the interface.
- When configuring IP source guard on interfaces on which a private VLAN is configured, port security is not supported.
- IP source guard is not supported on EtherChannels.
- You can enable this feature when IEEE 802.1x port-based authentication is enabled.
- If the number of ternary content addressable memory (TCAM) entries exceeds the maximum available, the CPU usage increases.

Enabling IP Source Guard

Beginning in privileged EXEC mode, follow these steps to enable and configure IP source guard on an interface.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled and NNIs are enabled.
Step 4	**ip verify source** or **ip verify source port-security**	Enable IP source guard with source IP address filtering. Enable IP source guard with source IP and MAC address filtering.

	Command	Purpose
Step 5	**exit**	Return to global configuration mode.
Step 6	**ip source binding** *mac-address* **vlan** *vlan-id ip-address* **inteface** *interface-id*	Add a static IP source binding. Enter this command for each static binding.
Step 7	**end**	Return to privileged EXEC mode.
Step 8	**show ip verify source** [**interface** *interface-id*]	Display the IP source guard configuration for all interfaces or for a specific interface.
Step 9	**show ip source binding** [*ip-address*] [*mac-address*] [**dhcp-snooping** \| **static**] [**inteface** *interface-id*] [**vlan** *vlan-id*]	Display the IP source bindings on the switch, on a specific VLAN, or on a specific interface.
Step 10	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable IP source guard with source IP address filtering, use the **no ip verify source** interface configuration command.

To delete a static IP source binding entry, use the **no ip source** global configuration command.

This example shows how to enable IP source guard with source IP and MAC filtering on VLANs 10 and 11:

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# ip verify source port-security
Switch(config-if)# exit
Switch(config)# ip source binding 0100.0022.0010 vlan 10 10.0.0.2 interface
gigabitethernet1/0/1
Switch(config)# ip source binding 0100.0230.0002 vlan 11 10.0.0.4 interface
gigabitethernet1/0/1
Switch(config)# end
```

Displaying IP Source Guard Information

To display the IP source guard information, use one or more of the privileged EXEC commands in Table 18-3:

Table 18-3 Commands for Displaying IP Source Guard Information

Command	Purpose
show ip source binding	Display the IP source bindings on a switch.
show ip verify source	Display the IP source guard configuration on the switch.



CHAPTER 19

Configuring Dynamic ARP Inspection

This chapter describes how to configure dynamic Address Resolution Protocol inspection (dynamic ARP inspection) on the Cisco ME 3400 switch. This feature helps prevent malicious attacks on the switch by not relaying invalid ARP requests and responses to other ports in the same VLAN.

Note This feature is supported only when the metro IP access or metro access image is running on the switch.

For complete syntax and usage information for the commands used in this chapter, see the command reference for this release.

This chapter consists of these sections:

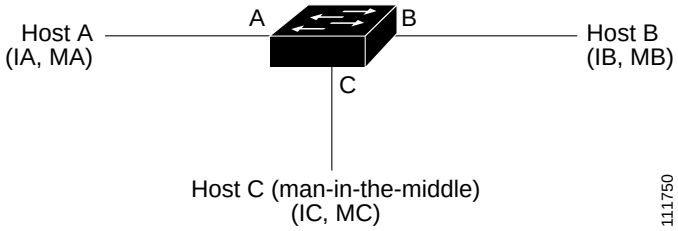
- Understanding Dynamic ARP Inspection, page 19-1
- Configuring Dynamic ARP Inspection, page 19-5
- Displaying Dynamic ARP Inspection Information, page 19-14

Understanding Dynamic ARP Inspection

ARP provides IP communication within a Layer 2 broadcast domain by mapping an IP address to a MAC address. For example, Host B wants to send information to Host A but does not have the MAC address of Host A in its ARP cache. Host B generates a broadcast message for all hosts within the broadcast domain to obtain the MAC address associated with the IP address of Host A. All hosts within the broadcast domain receive the ARP request, and Host A responds with its MAC address. However, because ARP allows a gratuitous reply from a host even if an ARP request was not received, an ARP spoofing attack and the poisoning of ARP caches can occur. After the attack, all traffic from the device under attack flows through the attacker's computer and then to the router, switch, or host.

A malicious user can attack hosts, switches, and routers connected to your Layer 2 network by poisoning the ARP caches of systems connected to the subnet and by intercepting traffic intended for other hosts on the subnet. Figure 19-1 shows an example of ARP cache poisoning.

Figure 19-1 ARP Cache Poisoning



Hosts A, B, and C are connected to the switch on interfaces A, B and C, all of which are on the same subnet. Their IP and MAC addresses are shown in parentheses; for example, Host A uses IP address IA and MAC address MA. When Host A needs to communicate to Host B at the IP layer, it broadcasts an ARP request for the MAC address associated with IP address IB. When the switch and Host B receive the ARP request, they populate their ARP caches with an ARP binding for a host with the IP address IA and a MAC address MA; for example, IP address IA is bound to MAC address MA. When Host B responds, the switch and Host A populate their ARP caches with a binding for a host with the IP address IB and the MAC address MB.

Host C can poison the ARP caches of the switch, Host A, and Host B by broadcasting forged ARP responses with bindings for a host with an IP address of IA (or IB) and a MAC address of MC. Hosts with poisoned ARP caches use the MAC address MC as the destination MAC address for traffic intended for IA or IB. This means that Host C intercepts that traffic. Because Host C knows the true MAC addresses associated with IA and IB, it can forward the intercepted traffic to those hosts by using the correct MAC address as the destination. Host C has inserted itself into the traffic stream from Host A to Host B, the classic *man-in-the middle* attack.

Dynamic ARP inspection is a security feature that validates ARP packets in a network. It intercepts, logs, and discards ARP packets with invalid IP-to-MAC address bindings. This capability protects the network from certain man-in-the-middle attacks.

Dynamic ARP inspection ensures that only valid ARP requests and responses are relayed. The switch performs these activities:

- Intercepts all ARP requests and responses on untrusted ports
- Verifies that each of these intercepted packets has a valid IP-to-MAC address binding before updating the local ARP cache or before forwarding the packet to the appropriate destination
- Drops invalid ARP packets

Dynamic ARP inspection determines the validity of an ARP packet based on valid IP-to-MAC address bindings stored in a trusted database, the DHCP snooping binding database. This database is built by DHCP snooping if DHCP snooping is enabled on the VLANs and on the switch. If the ARP packet is received on a trusted interface, the switch forwards the packet without any checks. On untrusted interfaces, the switch forwards the packet only if it is valid.

You enable dynamic ARP inspection on a per-VLAN basis by using the **ip arp inspection vlan** *vlan-range* global configuration command. For configuration information, see the "Configuring Dynamic ARP Inspection in DHCP Environments" section on page 19-7.

In non-DHCP environments, dynamic ARP inspection can validate ARP packets against user-configured ARP access control lists (ACLs) for hosts with statically configured IP addresses. You define an ARP ACL by using the **arp access-list** *acl-name* global configuration command. For configuration information, see the "Configuring ARP ACLs for Non-DHCP Environments" section on page 19-8. The switch logs dropped packets. For more information about the log buffer, see the "Logging of Dropped Packets" section on page 19-4.

You can configure dynamic ARP inspection to drop ARP packets when the IP addresses in the packets are invalid or when the MAC addresses in the body of the ARP packets do not match the addresses specified in the Ethernet header. Use the **ip arp inspection validate** {[**src-mac**] [**dst-mac**] [**ip**]} global configuration command. For more information, see the "Performing Validation Checks" section on page 19-12.

Interface Trust States and Network Security

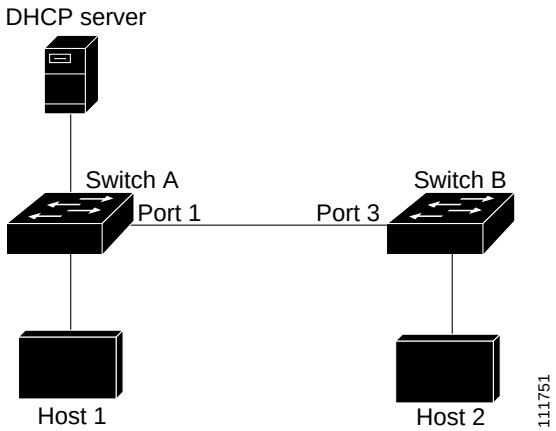
Dynamic ARP inspection associates a trust state with each interface on the switch. Packets arriving on trusted interfaces bypass all dynamic ARP inspection validation checks, and those arriving on untrusted interfaces undergo the dynamic ARP inspection validation process.

In a typical network configuration, you configure all switch ports connected to host ports as untrusted and configure all switch ports connected to switches as trusted. With this configuration, all ARP packets entering the network from a given switch bypass the security check. No other validation is needed at any other place in the VLAN or in the network. You configure the trust setting by using the **ip arp inspection trust** interface configuration command.

Caution Use the trust state configuration carefully. Configuring interfaces as untrusted when they should be trusted can result in a loss of connectivity.

In Figure 19-2, assume that both Switch A and Switch B are running dynamic ARP inspection on the VLAN that includes Host 1 and Host 2. If Host 1 and Host 2 acquire their IP addresses from the DHCP server connected to Switch A, only Switch A binds the IP-to-MAC address of Host 1. Therefore, if the interface between Switch A and Switch B is untrusted, the ARP packets from Host 1 are dropped by Switch B. Connectivity between Host 1 and Host 2 is lost.

Figure 19-2 ARP Packet Validation on a VLAN Enabled for Dynamic ARP Inspection



Configuring interfaces to be trusted when they are actually untrusted leaves a security hole in the network. If Switch A is not running dynamic ARP inspection, Host 1 can easily poison the ARP cache of Switch B (and Host 2, if the link between the switches is configured as trusted). This condition can occur even though Switch B is running dynamic ARP inspection.

Dynamic ARP inspection ensures that hosts (on untrusted interfaces) connected to a switch running dynamic ARP inspection do not poison the ARP caches of other hosts in the network. However, dynamic ARP inspection does not prevent hosts in other portions of the network from poisoning the caches of the hosts that are connected to a switch running dynamic ARP inspection.

In cases in which some switches in a VLAN run dynamic ARP inspection and other switches do not, configure the interfaces connecting such switches as untrusted. However, to validate the bindings of packets from nondynamic ARP inspection switches, configure the switch running dynamic ARP inspection with ARP ACLs. When you cannot determine such bindings, at Layer 3, isolate switches running dynamic ARP inspection from switches not running dynamic ARP inspection switches. For configuration information, see the "Configuring ARP ACLs for Non-DHCP Environments" section on page 19-8.

Note Depending on the setup of the DHCP server and the network, it might not be possible to validate a given ARP packet on all switches in the VLAN.

Rate Limiting of ARP Packets

The switch CPU performs dynamic ARP inspection validation checks; therefore, the number of incoming ARP packets is rate-limited to prevent a denial-of-service attack. By default, the rate for untrusted interfaces is 15 packets per second (pps). Trusted interfaces are not rate-limited. You can change this setting by using the **ip arp inspection limit** interface configuration command.

When the rate of incoming ARP packets exceeds the configured limit, the switch places the port in the error-disabled state. The port remains in that state until you intervene. You can use the **errdisable recovery** global configuration command to enable error disable recovery so that ports automatically emerge from this state after a specified timeout period.

For configuration information, see the "Limiting the Rate of Incoming ARP Packets" section on page 19-10.

Relative Priority of ARP ACLs and DHCP Snooping Entries

Dynamic ARP inspection uses the DHCP snooping binding database for the list of valid IP-to-MAC address bindings.

ARP ACLs take precedence over entries in the DHCP snooping binding database. The switch uses ACLs only if you configure them by using the **ip arp inspection filter vlan** global configuration command. The switch first compares ARP packets to user-configured ARP ACLs. If the ARP ACL denies the ARP packet, the switch also denies the packet even if a valid binding exists in the database populated by DHCP snooping.

Logging of Dropped Packets

When the switch drops a packet, it places an entry in the log buffer and then generates system messages on a rate-controlled basis. After the message is generated, the switch clears the entry from the log buffer. Each log entry contains flow information, such as the receiving VLAN, the port number, the source and destination IP addresses, and the source and destination MAC addresses.

You use the **ip arp inspection log-buffer** global configuration command to configure the number of entries in the buffer and the number of entries needed in the specified interval to generate system messages. You specify the type of packets that are logged by using the **ip arp inspection vlan logging** global configuration command. For configuration information, see the "Configuring the Log Buffer" section on page 19-13.

Configuring Dynamic ARP Inspection

These sections contain this configuration information:

- Default Dynamic ARP Inspection Configuration, page 19-5
- Dynamic ARP Inspection Configuration Guidelines, page 19-6
- Configuring Dynamic ARP Inspection in DHCP Environments, page 19-7 (required in DHCP environments)
- Configuring ARP ACLs for Non-DHCP Environments, page 19-8 (required in non-DHCP environments)
- Limiting the Rate of Incoming ARP Packets, page 19-10 (optional)
- Performing Validation Checks, page 19-12 (optional)
- Configuring the Log Buffer, page 19-13 (optional)

Default Dynamic ARP Inspection Configuration

Table 19-1 shows the default dynamic ARP inspection configuration.

Table 19-1 Default Dynamic ARP Inspection Configuration

Feature	Default Setting
Dynamic ARP inspection	Disabled on all VLANs.
Interface trust state	All interfaces are untrusted.
Rate limit of incoming ARP packets	The rate is 15 pps on untrusted interfaces, assuming that the network is a switched network with a host connecting to as many as 15 new hosts per second. The rate is unlimited on all trusted interfaces. The burst interval is 1 second.
ARP ACLs for non-DHCP environments	No ARP ACLs are defined.
Validation checks	No checks are performed.
Log buffer	When dynamic ARP inspection is enabled, all denied or dropped ARP packets are logged. The number of entries in the log is 32. The number of system messages is limited to 5 per second. The logging-rate interval is 1 second.
Per-VLAN logging	All denied or dropped ARP packets are logged.

Dynamic ARP Inspection Configuration Guidelines

These are the dynamic ARP inspection configuration guidelines:

Note This feature is supported only when the metro IP access or metro access image is running on the switch.

- Dynamic ARP inspection is an ingress security feature; it does not perform any egress checking.
- Dynamic ARP inspection is not effective for hosts connected to switches that do not support dynamic ARP inspection or that do not have this feature enabled. Because man-in-the-middle attacks are limited to a single Layer 2 broadcast domain, separate the domain with dynamic ARP inspection checks from the one with no checking. This action secures the ARP caches of hosts in the domain enabled for dynamic ARP inspection.
- Dynamic ARP inspection depends on the entries in the DHCP snooping binding database to verify IP-to-MAC address bindings in incoming ARP requests and ARP responses. Make sure to enable DHCP snooping to permit ARP packets that have dynamically assigned IP addresses. For configuration information, see Chapter 18, “Configuring DHCP Features and IP Source Guard.”

 When DHCP snooping is disabled or in non-DHCP environments, use ARP ACLs to permit or to deny packets.
- Dynamic ARP inspection is supported on access ports, trunk ports, EtherChannel ports, and private VLAN ports.
- A physical port can join an EtherChannel port channel only when the trust state of the physical port and the channel port match. Otherwise, the physical port remains suspended in the port channel. A port channel inherits its trust state from the first physical port that joins the channel. Consequently, the trust state of the first physical port need not match the trust state of the channel.

 Conversely, when you change the trust state on the port channel, the switch configures a new trust state on all the physical ports that comprise the channel.
- The operating rate for the port channel is cumulative across all the physical ports within the channel. For example, if you configure the port channel with an ARP rate-limit of 400 pps, all the interfaces combined on the channel receive an aggregate 400 pps. The rate of incoming ARP packets on EtherChannel ports is equal to the sum of the incoming rate of packets from all the channel members. Configure the rate limit for EtherChannel ports only after examining the rate of incoming ARP packets on the channel-port members.

 The rate of incoming packets on a physical port is checked against the port-channel configuration rather than the physical-ports configuration. The rate-limit configuration on a port channel is independent of the configuration on its physical ports.

 If the EtherChannel receives more ARP packets than the configured rate, the channel (including all physical ports) is placed in the error-disabled state.
- Make sure to limit the rate of ARP packets on incoming trunk ports. Configure trunk ports with higher rates to reflect their aggregation and to handle packets across multiple dynamic ARP inspection-enabled VLANs. You also can use the **ip arp inspection limit none** interface configuration command to make the rate unlimited. A high rate-limit on one VLAN can cause a denial-of-service attack to other VLANs when the software places the port in the error-disabled state.

Configuring Dynamic ARP Inspection in DHCP Environments

This procedure shows how to configure dynamic ARP inspection when two switches support this feature. Host 1 is connected to Switch A, and Host 2 is connected to Switch B as shown in Figure 19-2 on page 19-3. Both switches are running dynamic ARP inspection on VLAN 1 where the hosts are located. A DHCP server is connected to Switch A. Both hosts acquire their IP addresses from the same DHCP server. Therefore, Switch A has the bindings for Host 1 and Host 2, and Switch B has the binding for Host 2.

Note Dynamic ARP inspection depends on the entries in the DHCP snooping binding database to verify IP-to-MAC address bindings in incoming ARP requests and ARP responses. Make sure to enable DHCP snooping to permit ARP packets that have dynamically assigned IP addresses. For configuration information, see Chapter 18, "Configuring DHCP Features and IP Source Guard."

For information on how to configure dynamic ARP inspection when only one switch supports the feature, see the "Configuring ARP ACLs for Non-DHCP Environments" section on page 19-8.

Beginning in privileged EXEC mode, follow these steps to configure dynamic ARP inspection. You must perform this procedure on both switches. This procedure is required.

	Command	Purpose
Step 1	**show cdp neighbors**	Verify the connection between the switches.
Step 2	**configure terminal**	Enter global configuration mode.
Step 3	**ip arp inspection *vlan*** *vlan-range*	Enable dynamic ARP inspection on a per-VLAN basis. By default, dynamic ARP inspection is disabled on all VLANs. For *vlan-range*, specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. Specify the same VLAN ID for both switches.
Step 4	**interface** *interface-id*	Specify the interface connected to the other switch, and enter interface configuration mode.
Step 5	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled, and NNIs are enabled.
Step 6	**ip arp inspection trust**	Configure the connection between the switches as trusted. By default, all interfaces are untrusted. The switch does not check ARP packets that it receives from the other switch on the trusted interface. It simply forwards the packets. For untrusted interfaces, the switch intercepts all ARP requests and responses. It verifies that the intercepted packets have valid IP-to-MAC address bindings before updating the local cache and before forwarding the packet to the appropriate destination. The switch drops invalid packets and logs them in the log buffer according to the logging configuration specified with the **ip arp inspection vlan logging** global configuration command. For more information, see the "Configuring the Log Buffer" section on page 19-13.

	Command	Purpose
Step 7	**end**	Return to privileged EXEC mode.
Step 8	**show ip arp inspection interfaces** **show ip arp inspection vlan** *vlan-range*	Verify the dynamic ARP inspection configuration.
Step 9	**show ip dhcp snooping binding**	Verify the DHCP bindings.
Step 10	**show ip arp inspection statistics vlan** *vlan-range*	Check the dynamic ARP inspection statistics.
Step 11	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable dynamic ARP inspection, use the **no ip arp inspection vlan** *vlan-range* global configuration command. To return the interfaces to an untrusted state, use the **no ip arp inspection trust** interface configuration command.

This example shows how to configure dynamic ARP inspection on Switch A in VLAN 1. You would perform a similar procedure on Switch B:

```
Switch(config)# ip arp inspection vlan 1
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ip arp inspection trust
```

Configuring ARP ACLs for Non-DHCP Environments

This procedure shows how to configure dynamic ARP inspection when Switch B shown in Figure 19-2 on page 19-3 does not support dynamic ARP inspection or DHCP snooping.

If you configure port 1 on Switch A as trusted, a security hole is created because both Switch A and Host 1 could be attacked by either Switch B or Host 2. To prevent this possibility, you must configure port 1 on Switch A as untrusted. To permit ARP packets from Host 2, you must set up an ARP ACL and apply it to VLAN 1. If the IP address of Host 2 is not static (it is impossible to apply the ACL configuration on Switch A) you must separate Switch A from Switch B at Layer 3 and use a router to route packets between them.

Beginning in privileged EXEC mode, follow these steps to configure an ARP ACL on Switch A. This procedure is required in non-DHCP environments.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**arp access-list** *acl-name*	Define an ARP ACL, and enter ARP access-list configuration mode. By default, no ARP access lists are defined. **Note** At the end of the ARP access list, there is an implicit **deny ip any mac any** command.

	Command	Purpose
Step 3	**permit ip host** *sender-ip* **mac host** *sender-mac* [**log**]	Permit ARP packets from the specified host (Host 2). • For *sender-ip*, enter the IP address of Host 2. • For *sender-mac*, enter the MAC address of Host 2. • (Optional) Specify **log** to log a packet in the log buffer when it matches the access control entry (ACE). Matches are logged if you also configure the **matchlog** keyword in the **ip arp inspection vlan logging** global configuration command. For more information, see the "Configuring the Log Buffer" section on page 19-13.
Step 4	**exit**	Return to global configuration mode.
Step 5	**ip arp inspection filter** *arp-acl-name* **vlan** *vlan-range* [**static**]	Apply the ARP ACL to the VLAN. By default, no defined ARP ACLs are applied to any VLAN. • For *arp-acl-name*, specify the name of the ACL created in Step 2. • For *vlan-range*, specify the VLAN that the switches and hosts are in. You can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. • (Optional) Specify **static** to treat implicit denies in the ARP ACL as explicit denies and to drop packets that do not match any previous clauses in the ACL. DHCP bindings are not used. If you do not specify this keyword, it means that there is no explicit deny in the ACL that denies the packet, and DHCP bindings determine whether a packet is permitted or denied if the packet does not match any clauses in the ACL. ARP packets containing only IP-to-MAC address bindings are compared against the ACL. Packets are permitted only if the access list permits them.
Step 6	**interface** *interface-id*	Specify the Switch A interface that is connected to Switch B, and enter interface configuration mode.
Step 7	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled, and NNIs are enabled.

	Command	Purpose
Step 8	**no ip arp inspection trust**	Configure the Switch A interface that is connected to Switch B as untrusted. By default, all interfaces are untrusted. For untrusted interfaces, the switch intercepts all ARP requests and responses. It verifies that the intercepted packets have valid IP-to-MAC address bindings before updating the local cache and before forwarding the packet to the appropriate destination. The switch drops invalid packets and logs them in the log buffer according to the logging configuration specified with the **ip arp inspection vlan logging** global configuration command. For more information, see the "Configuring the Log Buffer" section on page 19-13.
Step 9	**end**	Return to privileged EXEC mode.
Step 10	**show arp access-list** [*acl-name*] **show ip arp inspection vlan** *vlan-range* **show ip arp inspection interfaces**	Verify your entries.
Step 11	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the ARP ACL, use the **no arp access-list** global configuration command. To remove the ARP ACL attached to a VLAN, use the **no ip arp inspection filter** *arp-acl-name* **vlan** *vlan-range* global configuration command.

This example shows how to configure an ARP ACL called *host2* on Switch A, to permit ARP packets from Host 2 (IP address 1.1.1.1 and MAC address 0001.0001.0001), to apply the ACL to VLAN 1, and to configure port 1 on Switch A as untrusted:

```
Switch(config)# arp access-list host2
Switch(config-arp-acl)# permit ip host 1.1.1.1 mac host 1.1.1
Switch(config-arp-acl)# exit
Switch(config)# ip arp inspection filter host2 vlan 1
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# no ip arp inspection trust
```

Limiting the Rate of Incoming ARP Packets

The switch CPU performs dynamic ARP inspection validation checks; therefore, the number of incoming ARP packets is rate-limited to prevent a denial-of-service attack.

When the rate of incoming ARP packets exceeds the configured limit, the switch places the port in the error-disabled state. The port remains in that state until you enable error-disabled recovery so that ports automatically emerge from this state after a specified timeout period.

Note Unless you configure a rate limit on an interface, changing the trust state of the interface also changes its rate limit to the default value for that trust state. After you configure the rate limit, the interface retains the rate limit even when its trust state is changed. If you enter the **no ip arp inspection limit** interface configuration command, the interface reverts to its default rate limit.

For configuration guidelines for rate limiting trunk ports and EtherChannel ports, see the "Dynamic ARP Inspection Configuration Guidelines" section on page 19-6.

Beginning in privileged EXEC mode, follow these steps to limit the rate of incoming ARP packets. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to be rate-limited, and enter interface configuration mode.
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled, and NNIs are enabled.
Step 4	**ip arp inspection limit** {**rate** *pps* [**burst interval** *seconds*] \| **none**}	Limit the rate of incoming ARP requests and responses on the interface. The default rate is 15 pps on untrusted interfaces and unlimited on trusted interfaces. The burst interval is 1 second. The keywords have these meanings: • For **rate** *pps*, specify an upper limit for the number of incoming packets processed per second. The range is 0 to 2048 pps. • (Optional) For **burst interval** *seconds*, specify the consecutive interval in seconds, over which the interface is monitored for a high rate of ARP packets.The range is 1 to 15. • For **rate none**, specify no upper limit for the rate of incoming ARP packets that can be processed.
Step 5	**exit**	Return to global configuration mode.
Step 6	**errdisable recovery cause arp-inspection interval** *interval*	(Optional) Enable error recovery from the dynamic ARP inspection error-disable state. By default, recovery is disabled, and the recovery interval is 300 seconds. For **interval** *interval*, specify the time in seconds to recover from the error-disable state. The range is 30 to 86400.
Step 7	**exit**	Return to privileged EXEC mode.
Step 8	**show ip arp inspection interfaces** **show errdisable recovery**	Verify your settings.
Step 9	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default rate-limit configuration, use the **no ip arp inspection limit** interface configuration command. To disable error recovery for dynamic ARP inspection, use the **no errdisable recovery cause arp-inspection** global configuration command.

Performing Validation Checks

Dynamic ARP inspection intercepts, logs, and discards ARP packets with invalid IP-to-MAC address bindings. You can configure the switch to perform additional checks on the destination MAC address, the sender and target IP addresses, and the source MAC address.

Beginning in privileged EXEC mode, follow these steps to perform specific checks on incoming ARP packets. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip arp inspection validate** {[**src-mac**] [**dst-mac**] [**ip**]}	Perform a specific check on incoming ARP packets. By default, no checks are performed. The keywords have these meanings: • For **src-mac**, check the source MAC address in the Ethernet header against the sender MAC address in the ARP body. This check is performed on both ARP requests and responses. When enabled, packets with different MAC addresses are classified as invalid and are dropped. • For **dst-mac**, check the destination MAC address in the Ethernet header against the target MAC address in ARP body. This check is performed for ARP responses. When enabled, packets with different MAC addresses are classified as invalid and are dropped. • For **ip**, check the ARP body for invalid and unexpected IP addresses. Addresses include 0.0.0.0, 255.255.255.255, and all IP multicast addresses. Sender IP addresses are checked in all ARP requests and responses, and target IP addresses are checked only in ARP responses. You must specify at least one of the keywords. Each command overrides the configuration of the previous command; that is, if a command enables **src** and **dst mac** validations, and a second command enables IP validation only, the **src** and **dst mac** validations are disabled as a result of the second command.
Step 3	**exit**	Return to privileged EXEC mode.
Step 4	**show ip arp inspection vlan** *vlan-range*	Verify your settings.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable checking, use the **no ip arp inspection validate** [**src-mac**] [**dst-mac**] [**ip**] global configuration command. To display statistics for forwarded, dropped, and MAC and IP validation failure packets, use the **show ip arp inspection statistics** privileged EXEC command.

Configuring the Log Buffer

When the switch drops a packet, it places an entry in the log buffer and then generates system messages on a rate-controlled basis. After the message is generated, the switch clears the entry from the log buffer. Each log entry contains flow information, such as the receiving VLAN, the port number, the source and destination IP addresses, and the source and destination MAC addresses.

A log-buffer entry can represent more than one packet. For example, if an interface receives many packets on the same VLAN with the same ARP parameters, the switch combines the packets as one entry in the log buffer and generates a single system message for the entry.

If the log buffer overflows, it means that a log event does not fit into the log buffer, and the display for the **show ip arp inspection log** privileged EXEC command is affected. A -- in the display appears in place of all data except the packet count and the time. No other statistics are provided for the entry. If you see this entry in the display, increase the number of entries in the log buffer or increase the logging rate.

Beginning in privileged EXEC mode, follow these steps to configure the log buffer. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip arp inspection log-buffer** {**entries** *number* \| **logs** *number* **interval** *seconds*}	Configure the dynamic ARP inspection logging buffer. By default, when dynamic ARP inspection is enabled, denied or dropped ARP packets are logged. The number of log entries is 32. The number of system messages is limited to 5 per second. The logging-rate interval is 1 second. The keywords have these meanings: • For **entries** *number*, specify the number of entries to be logged in the buffer. The range is 0 to 1024. • For **logs** *number* **interval** *seconds*, specify the number of entries to generate system messages in the specified interval. For **logs** *number*, the range is 0 to 1024. A 0 value means that the entry is placed in the log buffer, but a system message is not generated. For **interval** *seconds*, the range is 0 to 86400 seconds (1 day). A 0 value means that a system message is immediately generated (and the log buffer is always empty). An interval setting of 0 overrides a log setting of 0. The **logs** and **interval** settings interact. If the **logs** *number* X is greater than **interval** *seconds* Y, X divided by Y (X/Y) system messages are sent every second. Otherwise, one system message is sent every Y divided by X (Y/X) seconds.

	Command	Purpose
Step 3	**ip arp inspection vlan** *vlan-range* **logging** {**acl-match** {**matchlog** \| **none**} \| **dhcp-bindings** {**all** \| **none** \| **permit**}}	Control the type of packets that are logged per VLAN. By default, all denied or all dropped packets are logged. The term *logged* means the entry is placed in the log buffer and a system message is generated. The keywords have these meanings: • For *vlan-range*, specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. • For **acl-match matchlog**, log packets based on the ACE logging configuration. If you specify the **matchlog** keyword in this command and the **log** keyword in the **permit** or **deny** ARP access-list configuration command, ARP packets permitted or denied by the ACL are logged. • For **acl-match none**, do not log packets that match ACLs. • For **dhcp-bindings all**, log all packets that match DHCP bindings. • For **dhcp-bindings none**, do not log packets that match DHCP bindings. • For **dhcp-bindings permit**, log DHCP-binding permitted packets.
Step 4	**exit**	Return to privileged EXEC mode.
Step 5	**show ip arp inspection log**	Verify your settings.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default log buffer settings, use the **no ip arp inspection log-buffer** {**entries** | **logs**} global configuration command. To return to the default VLAN log settings, use the **no ip arp inspection vlan** *vlan-range* **logging** {**acl-match** | **dhcp-bindings**} global configuration command. To clear the log buffer, use the **clear ip arp inspection log** privileged EXEC command.

Displaying Dynamic ARP Inspection Information

To display dynamic ARP inspection information, use the privileged EXEC commands described in Table 19-2:

Table 19-2 Commands for Displaying Dynamic ARP Inspection Information

Command	Description
show arp access-list [*acl-name*]	Displays detailed information about ARP ACLs.
show ip arp inspection interfaces [*interface-id*]	Displays the trust state and the rate limit of ARP packets for the specified interface or all interfaces.
show ip arp inspection vlan *vlan-range*	Displays the configuration and the operating state of dynamic ARP inspection for the specified VLAN. If no VLANs are specified or if a range is specified, displays information only for VLANs with dynamic ARP inspection enabled (active).

To clear or display dynamic ARP inspection statistics, use the privileged EXEC commands in Table 19-3:

Table 19-3 Commands for Clearing or Displaying Dynamic ARP Inspection Statistics

Command	Description
clear ip arp inspection statistics	Clears dynamic ARP inspection statistics.
show ip arp inspection statistics [**vlan** *vlan-range*]	Displays statistics for forwarded, dropped, MAC validation failure, IP validation failure, ACL permitted and denied, and DHCP permitted and denied packets for the specified VLAN. If no VLANs are specified or if a range is specified, displays information only for VLANs with dynamic ARP inspection enabled (active).

For the **show ip arp inspection statistics** command, the switch increments the number of forwarded packets for each ARP request and response packet on a trusted dynamic ARP inspection port. The switch increments the number of ACL or DHCP permitted packets for each packet that is denied by source MAC, destination MAC, or IP validation checks, and the switch increments the appropriate failure count.

To clear or display dynamic ARP inspection logging information, use the privileged EXEC commands in Table 19-4:

Table 19-4 Commands for Clearing or Displaying Dynamic ARP Inspection Logging Information

Command	Description
clear ip arp inspection log	Clears the dynamic ARP inspection log buffer.
show ip arp inspection log	Displays the configuration and contents of the dynamic ARP inspection log buffer.

For more information about these commands, see the command reference for this release.



Configuring IGMP Snooping and MVR

This chapter describes how to configure Internet Group Management Protocol (IGMP) snooping on the Cisco ME 3400 Ethernet Access switch, including an application of local IGMP snooping, Multicast VLAN Registration (MVR). It also includes procedures for controlling multicast group membership by using IGMP filtering and procedures for configuring the IGMP throttling action.

Note For complete syntax and usage information for the commands used in this chapter, see the switch command reference for this release and the "IP Multicast Routing Commands" section in the *Cisco IOS IP Command Reference, Volume 3 of 3:Multicast, Release 12.2*.

This chapter consists of these sections:

- Understanding IGMP Snooping, page 20-1
- Configuring IGMP Snooping, page 20-6
- Displaying IGMP Snooping Information, page 20-15
- Understanding Multicast VLAN Registration, page 20-16
- Configuring MVR, page 20-18
- Displaying MVR Information, page 20-22
- Configuring IGMP Filtering and Throttling, page 20-22
- Displaying IGMP Filtering and Throttling Configuration, page 20-28

Note You can either manage IP multicast group addresses through features such as IGMP snooping and MVR, or you can use static IP addresses.

Understanding IGMP Snooping

Layer 2 switches can use IGMP snooping to constrain the flooding of multicast traffic by dynamically configuring Layer 2 interfaces so that multicast traffic is forwarded to only those interfaces associated with IP multicast devices. As the name implies, IGMP snooping requires the LAN switch to snoop on the IGMP transmissions between the host and the router and to keep track of multicast groups and member ports. When the switch receives an IGMP report from a host for a particular multicast group, the switch adds the host port number to the forwarding table entry; when it receives an IGMP Leave Group message from a host, it removes the host port from the table entry. It also periodically deletes entries if it does not receive IGMP membership reports from the multicast clients.

Note For more information on IP multicast and IGMP, see RFC 1112 and RFC 2236.

The multicast router sends out periodic general queries to all VLANs. All hosts interested in this multicast traffic send join requests and are added to the forwarding table entry. The switch creates one entry per VLAN in the IGMP snooping IP multicast forwarding table for each group from which it receives an IGMP join request.

The switch supports IP multicast group-based bridging, rather than MAC-addressed based groups. With multicast MAC address-based groups, if an IP address being configured translates (aliases) to a previously configured MAC address or to any reserved multicast MAC addresses (in the range 224.0.0.xxx), the command fails. Because the switch uses IP multicast groups, there are no address aliasing issues.

The IP multicast groups learned through IGMP snooping are dynamic. However, you can statically configure multicast groups by using the **ip igmp snooping vlan** *vlan-id* **static** *ip_address* **interface** *interface-id* global configuration command. If you specify group membership for a multicast group address statically, your setting supersedes any automatic manipulation by IGMP snooping. Multicast group membership lists can consist of both user-defined and IGMP snooping-learned settings.

You can configure an IGMP snooping querier to support IGMP snooping in subnets without multicast interfaces because the multicast traffic does not need to be routed. For more information about the IGMP snooping querier, see the "Configuring the IGMP Snooping Querier" section on page 20-13.

If a port spanning-tree, a port group, or a VLAN ID change occurs, the IGMP snooping-learned multicast groups from this port on the VLAN are deleted.

These sections describe IGMP snooping characteristics:

- IGMP Versions, page 20-2
- Joining a Multicast Group, page 20-3
- Leaving a Multicast Group, page 20-5
- Immediate Leave, page 20-5
- IGMP Configurable-Leave Timer, page 20-5
- IGMP Report Suppression, page 20-6

IGMP Versions

The switch supports IGMP Version 1, IGMP Version 2, and IGMP Version 3. These versions are interoperable on the switch. For example, if IGMP snooping is enabled on an IGMPv2 switch and the switch receives an IGMPv3 report from a host, the switch can forward the IGMPv3 report to the multicast router.

Note The switches support IGMPv3 snooping based only on the destination multicast MAC address. They do not support snooping based on the source MAC address or on proxy reports.

An IGMPv3 switch supports Basic IGMPv3 Snooping Support (BISS), which includes support for the snooping features on IGMPv1 and IGMPv2 switches and for IGMPv3 membership report messages. BISS constrains the flooding of multicast traffic when your network includes IGMPv3 hosts. It constrains traffic to approximately the same set of ports as the IGMP snooping feature on IGMPv2 or IGMPv1 hosts.

Note IGMPv3 join and leave messages are not supported on switches running IGMP filtering or MVR.

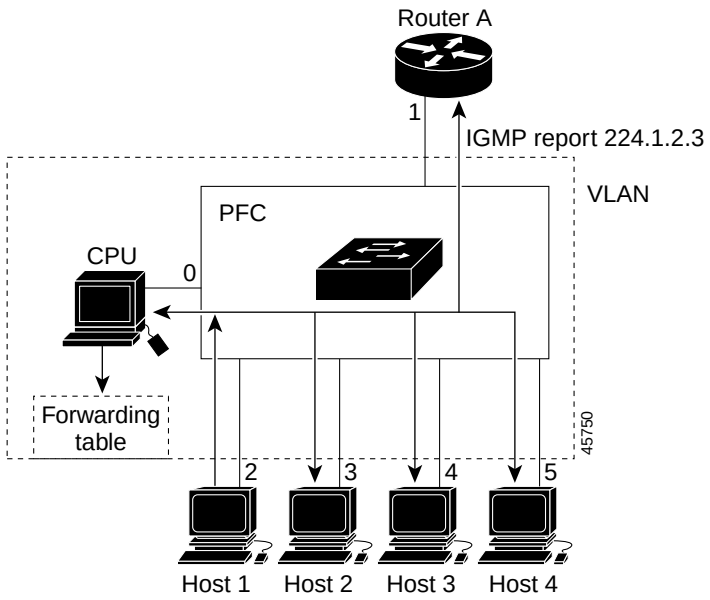
An IGMPv3 switch can receive messages from and forward messages to a device running the Source Specific Multicast (SSM) feature. For more information about source-specific multicast with IGMPv3 and IGMP, see this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios121/121newft/121t/121t5/dtssm5t.htm

Joining a Multicast Group

When a host connected to the switch wants to join an IP multicast group and it is an IGMP Version 2 client, it sends an unsolicited IGMP join message, specifying the IP multicast group to join. Alternatively, when the switch receives a general query from the router, it forwards the query to all ports in the VLAN. IGMP Version 1 or Version 2 hosts wanting to join the multicast group respond by sending a join message to the switch. The switch CPU creates a multicast forwarding-table entry for the group if it is not already present. The CPU also adds the interface where the join message was received to the forwarding-table entry. The host associated with that interface receives multicast traffic for that multicast group. See Figure 20-1.

Figure 20-1 Initial IGMP Join Message



Router A sends a general query to the switch, which forwards the query to ports 2 through 5, which are all members of the same VLAN. Host 1 wants to join multicast group 224.1.2.3 and multicasts an IGMP membership report (IGMP join message) to the group. The switch CPU uses the information in the IGMP report to set up a forwarding-table entry, as shown in Table 20-1, that includes the port numbers connected to Host 1 and the router.

Table 20-1 IGMP Snooping Forwarding Table

Destination Address	Type of Packet	Ports
224.1.2.3	IGMP	1, 2

The switch hardware can distinguish IGMP information packets from other packets for the multicast group. The information in the table tells the switching engine to send frames addressed to the 224.1.2.3 multicast IP address that are not IGMP packets to the router and to the host that has joined the group.

If another host (for example, Host 4) sends an unsolicited IGMP join message for the same group (Figure 20-2), the CPU receives that message and adds the port number of Host 4 to the forwarding table as shown in Table 20-2. Note that because the forwarding table directs IGMP messages to only the CPU, the message is not flooded to other ports on the switch. Any known multicast traffic is forwarded to the group and not to the CPU.

Figure 20-2 Second Host Joining a Multicast Group

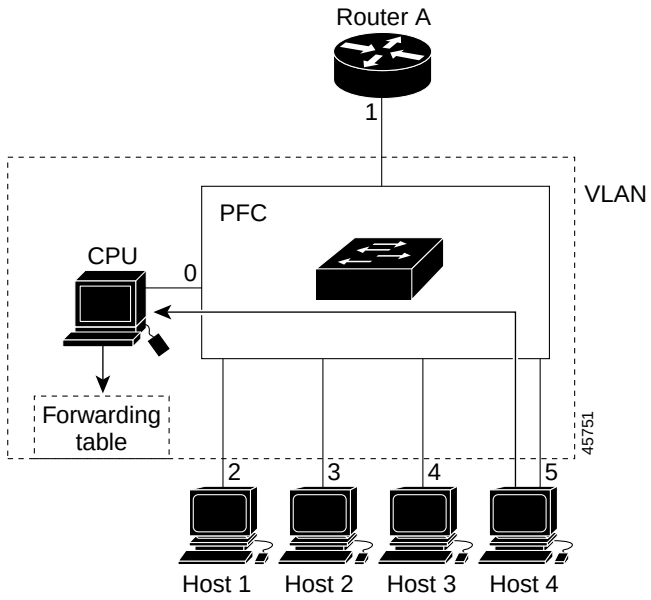


Table 20-2 Updated IGMP Snooping Forwarding Table

Destination Address	Type of Packet	Ports
224.1.2.3	IGMP	1, 2, 5

Multicast-capable router ports are added to the forwarding table for every Layer 2 multicast entry. The switch learns of such ports through one of these methods:

- Snooping on IGMP queries and Protocol Independent Multicast (PIM) packets
- Statically connecting to a multicast router port with the **ip igmp snooping mrouter** global configuration command

Leaving a Multicast Group

The router sends periodic multicast general queries, and the switch forwards these queries through all ports in the VLAN. Interested hosts respond to the queries. If at least one host in the VLAN wishes to receive multicast traffic, the router continues forwarding the multicast traffic to the VLAN. The switch forwards multicast group traffic only to those hosts listed in the forwarding table for that IP multicast group maintained by IGMP snooping.

When hosts want to leave a multicast group, they can silently leave, or they can send a leave message. When the switch receives a leave message from a host, it sends a group-specific query to learn if any other devices connected to that interface are interested in traffic for the specific multicast group. The switch then updates the forwarding table for that MAC group so that only those hosts interested in receiving multicast traffic for the group are listed in the forwarding table. If the router receives no reports from a VLAN, it removes the group for the VLAN from its IGMP cache.

Immediate Leave

Immediate Leave is only supported on IGMP Version 2 hosts.

The switch uses IGMP snooping Immediate Leave to remove from the forwarding table an interface that sends a leave message without the switch sending group-specific queries to the interface. The VLAN interface is pruned from the multicast tree for the multicast group specified in the original leave message. Immediate Leave ensures optimal bandwidth management for all hosts on a switched network, even when multiple multicast groups are simultaneously in use.

Note You should only use the Immediate Leave feature on VLANs where a single host is connected to each port. If Immediate Leave is enabled in VLANs where more than one host is connected to a port, some hosts might inadvertently be dropped.

For configuration steps, see the "Enabling IGMP Immediate Leave" section on page 20-9.

IGMP Configurable-Leave Timer

You can configure the time that the switch waits after sending a group-specific query to determine if hosts are still interested in a specific multicast group. The IGMP leave response time can be configured from 100 to 5000 milliseconds. The timer can be set either globally or on a per-VLAN basis. The VLAN configuration of the leave time overrides the global configuration.

For configuration steps, see the "Configuring the IGMP Leave Timer" section on page 20-10.

IGMP Report Suppression

Note IGMP report suppression is supported only when the multicast query has IGMPv1 and IGMPv2 reports. This feature is not supported when the query includes IGMPv3 reports.

The switch uses IGMP report suppression to forward only one IGMP report per multicast router query to multicast devices. When IGMP router suppression is enabled (the default), the switch sends the first IGMP report from all hosts for a group to all the multicast routers. The switch does not send the remaining IGMP reports for the group to the multicast routers. This feature prevents duplicate reports from being sent to the multicast devices.

If the multicast router query includes requests only for IGMPv1 and IGMPv2 reports, the switch forwards only the first IGMPv1 or IGMPv2 report from all hosts for a group to all the multicast routers.

If the multicast router query also includes requests for IGMPv3 reports, the switch forwards all IGMPv1, IGMPv2, and IGMPv3 reports for a group to the multicast devices.

If you disable IGMP report suppression, all IGMP reports are forwarded to the multicast routers. For configuration steps, see the "Disabling IGMP Report Suppression" section on page 20-14.

Configuring IGMP Snooping

IGMP snooping allows switches to examine IGMP packets and make forwarding decisions based on their content. These sections contain this configuration information:

- Default IGMP Snooping Configuration, page 20-6
- Enabling or Disabling IGMP Snooping, page 20-7
- Configuring a Multicast Router Port, page 20-8
- Configuring a Host Statically to Join a Group, page 20-8
- Enabling IGMP Immediate Leave, page 20-9
- Configuring the IGMP Leave Timer, page 20-10
- Configuring TCN-Related Commands, page 20-11
- Configuring the IGMP Snooping Querier, page 20-13
- Disabling IGMP Report Suppression, page 20-14

Default IGMP Snooping Configuration

Table 20-3 shows the default IGMP snooping configuration.

Table 20-3 Default IGMP Snooping Configuration

Feature	Default Setting
IGMP snooping	Enabled globally and per VLAN
Multicast routers	None configured
Multicast router learning (snooping) method	PIM

Table 20-3 Default IGMP Snooping Configuration (continued)

Feature	Default Setting
IGMP snooping Immediate Leave	Disabled
Static groups	None configured
TCN[1] flood query count	2
TCN query solicitation	Disabled
IGMP snooping querier	Disabled
IGMP report suppression	Enabled

1. TCN = Topology Change Notification

Enabling or Disabling IGMP Snooping

By default, IGMP snooping is globally enabled on the switch. When globally enabled or disabled, it is also enabled or disabled in all existing VLAN interfaces. IGMP snooping is by default enabled on all VLANs, but can be enabled and disabled on a per-VLAN basis.

Global IGMP snooping overrides the VLAN IGMP snooping. If global snooping is disabled, you cannot enable VLAN snooping. If global snooping is enabled, you can enable or disable VLAN snooping.

Beginning in privileged EXEC mode, follow these steps to globally enable IGMP snooping on the switch:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip igmp snooping**	Globally enable IGMP snooping in all existing VLAN interfaces.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To globally disable IGMP snooping on all VLAN interfaces, use the **no ip igmp snooping** global configuration command.

Beginning in privileged EXEC mode, follow these steps to enable IGMP snooping on a VLAN interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip igmp snooping vlan** *vlan-id*	Enable IGMP snooping on the VLAN interface.The VLAN ID range is 1 to 1001 and 1006 to 4094. **Note** IGMP snooping must be globally enabled before you can enable VLAN snooping.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable IGMP snooping on a VLAN interface, use the **no ip igmp snooping vlan** *vlan-id* global configuration command for the specified VLAN number.

Configuring a Multicast Router Port

To add a multicast router port (add a static connection to a multicast router), use the **ip igmp snooping vlan mrouter** global configuration command on the switch.

Note Static connections to multicast routers are supported only on switch ports.

Beginning in privileged EXEC mode, follow these steps to enable a static connection to a multicast router:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip igmp snooping vlan** *vlan-id* **mrouter interface** *interface-id*	Specify the multicast router VLAN ID and the interface to the multicast router. • The VLAN ID range is 1 to 1001 and 1006 to 4094. • The interface can be a physical interface or a port channel. The port-channel range is 1 to 48.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show ip igmp snooping mrouter** [**vlan** *vlan-id*]	Verify that IGMP snooping is enabled on the VLAN interface.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove a multicast router port from the VLAN, use the **no ip igmp snooping vlan** *vlan-id* **mrouter interface** *interface-id* global configuration command.

This example shows how to enable a static connection to a multicast router:

```
Switch# configure terminal
Switch(config)# ip igmp snooping vlan 200 mrouter interface gigabitethernet0/2
Switch(config)# end
```

Configuring a Host Statically to Join a Group

Hosts or Layer 2 ports normally join multicast groups dynamically, but you can also statically configure a host on an interface.

Beginning in privileged EXEC mode, follow these steps to add a Layer 2 port as a member of a multicast group:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip igmp snooping vlan** *vlan-id* **static** *ip_address* **interface** *interface-id*	Statically configure a Layer 2 port as a member of a multicast group: • *vlan-id* is the multicast group VLAN ID. The range is 1 to 1001 and 1006 to 4094. • *ip-address* is the group IP address. • *interface-id* is the member port. It can be a physical interface or a port channel (1 to 48).
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show ip igmp snooping groups**	Verify the member port and the IP address.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the Layer 2 port from the multicast group, use the **no ip igmp snooping vlan** *vlan-id* **static** *mac-address* **interface** *interface-id* global configuration command.

This example shows how to statically configure a host on a port:

```
Switch# configure terminal
Switch(config)# ip igmp snooping vlan 105 static 224.2.4.12 interface gigabitethernet0/1
Switch(config)# end
```

Enabling IGMP Immediate Leave

When you enable IGMP Immediate Leave, the switch immediately removes a port when it detects an IGMP Version 2 leave message on that port. You should only use the Immediate-Leave feature when there is a single receiver present on every port in the VLAN.

Note Immediate Leave is supported only on IGMP Version 2 hosts.

Beginning in privileged EXEC mode, follow these steps to enable IGMP Immediate Leave:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip igmp snooping vlan** *vlan-id* **immediate-leave**	Enable IGMP Immediate Leave on the VLAN interface.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show ip igmp snooping vlan** *vlan-id*	Verify that Immediate Leave is enabled on the VLAN interface.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable IGMP Immediate Leave on a VLAN, use the **no ip igmp snooping vlan** *vlan-id* **immediate-leave** global configuration command.

This example shows how to enable IGMP Immediate Leave on VLAN 130:

```
Switch# configure terminal
Switch(config)# ip igmp snooping vlan 130 immediate-leave
Switch(config)# end
```

Configuring the IGMP Leave Timer

Follow these guidelines when configuring the IGMP leave timer:

- You can configure the leave time globally or on a per-VLAN basis.
- Configuring the leave time on a VLAN overrides the global setting.
- The default leave time is 1000 milliseconds.
- The IGMP configurable leave time is only supported on hosts running IGMP Version 2.
- The actual leave latency in the network is usually the configured leave time. However, the leave time *might* vary around the configured time, depending on real-time CPU load conditions, network delays and the amount of traffic sent through the interface.

Beginning in privileged EXEC mode, follow these steps to enable the IGMP configurable-leave timer:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip igmp snooping last-member-query-interval** *time*	Configure the IGMP leave timer globally. The range is 100 to 5000 milliseconds. The default is 1000 seconds.
Step 3	**ip igmp snooping vlan** *vlan-id* **last-member-query-interval** *time*	(Optional) Configure the IGMP leave time on the VLAN interface. The range is 100 to 5000 milliseconds. **Note** Configuring the leave time on a VLAN overrides the globally configured timer.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show ip igmp snooping**	(Optional) Display the configured IGMP leave time.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no ip igmp snooping last-member-query-interva**l global configuration command to globally reset the IGMP leave timer to the default setting.

Use the **no ip igmp snooping vlan** *vlan-id* **last-member-query-interval** global configuration command to remove the configured IGMP leave-time setting from the specified VLAN.

Configuring TCN-Related Commands

These sections describe how to control flooded multicast traffic during a TCN event:

- Controlling the Multicast Flooding Time After a TCN Event, page 20-11
- Recovering from Flood Mode, page 20-11
- Disabling Multicast Flooding During a TCN Event, page 20-12

Controlling the Multicast Flooding Time After a TCN Event

You can control the time that multicast traffic is flooded after a TCH event by using the **ip igmp snooping tcn flood query count** global configuration command. This command configures the number of general queries for which multicast data traffic is flooded after a TCN event. Some examples of TCN events are the client changed its location and the receiver is on same port that was blocked but is now forwarding, and a port went down without sending a leave message.

If you set the TCN flood query count to 1 by using the **ip igmp snooping tcn flood query count** command, the flooding stops after receiving one general query. If you set the count to 7, the flooding of multicast traffic due to the TCN event lasts until 7 general queries are received. Groups are relearned based on the general queries received during the TCN event.

Beginning in privileged EXEC mode, follow these steps to configure the TCN flood query count:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip igmp snooping tcn flood query count** *count*	Specify the number of IGMP general queries for which the multicast traffic is flooded. The range is 1 to 10. By default, the flooding query count is 2.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show ip igmp snooping**	Verify the TCN settings.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default flooding query count, use the **no ip igmp snooping tcn flood query count** global configuration command.

Recovering from Flood Mode

When a topology change occurs, the spanning-tree root sends a special IGMP leave message (also known as global leave) with the group multicast address 0.0.0.0. However, when you enable the **ip igmp snooping tcn query solicit** global configuration command, the switch sends the global leave message whether or not it is the spanning-tree root. When the router receives this special leave, it immediately sends general queries, which expedite the process of recovering from the flood mode during the TCN event. Leaves are always sent if the switch is the spanning-tree root regardless of this configuration command. By default, query solicitation is disabled.

Beginning in privileged EXEC mode, follow these steps to enable the switch sends the global leave message whether or not it is the spanning-tree root:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip igmp snooping tcn query solicit**	Send an IGMP leave message (global leave) to speed the process of recovering from the flood mode caused during a TCN event. By default, query solicitation is disabled.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show ip igmp snooping**	Verify the TCN settings.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default query solicitation, use the **no ip igmp snooping tcn query solicit** global configuration command.

Disabling Multicast Flooding During a TCN Event

When the switch receives a TCN, multicast traffic is flooded to all the ports until two general queries are received. If the switch has many ports with attached hosts that are subscribed to different multicast groups, the flooding might exceed the capacity of the link and cause packet loss. You can use the **ip igmp snooping tcn flood** interface configuration command to control this behavior.

Beginning in privileged EXEC mode, follow these steps to disable multicast flooding on an interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
Step 3	**no shutdown**	Enable the port, if necessary. By default, user network interfaces (UNIs) are disabled, and network node interfaces (NNIs) are enabled.
Step 4	**no ip igmp snooping tcn flood**	Disable the flooding of multicast traffic during a spanning-tree TCN event. By default, multicast flooding is enabled on an interface.
Step 5	**exit**	Return to privileged EXEC mode.
Step 6	**show ip igmp snooping**	Verify the TCN settings.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To re-enable multicast flooding on an interface, use the **ip igmp snooping tcn flood** interface configuration command.

Configuring the IGMP Snooping Querier

Follow these guidelines when configuring the IGMP snooping querier:

- Configure the VLAN in global configuration mode.
- Configure an IP address on the VLAN interface. When enabled, the IGMP snooping querier uses the IP address as the query source address.
- If there is no IP address configured on the VLAN interface, the IGMP snooping querier tries to use the configured global IP address for the IGMP querier. If there is no global IP address specified, the IGMP querier tries to use the VLAN switch virtual interface (SVI) IP address (if one exists). If there is no SVI IP address, the switch uses the first available IP address configured on the switch. The first IP address available appears in the output of the **show ip interface** privileged EXEC command. The IGMP snooping querier does not generate an IGMP general query if it cannot find an available IP address on the switch.
- The IGMP snooping querier supports IGMP Versions 1 and 2.
- When administratively enabled, the IGMP snooping querier moves to the nonquerier state if it detects the presence of a multicast router in the network.
- When it is administratively enabled, the IGMP snooping querier moves to the operationally disabled state under these conditions:
 - IGMP snooping is disabled in the VLAN.
 - PIM is enabled on the SVI of the corresponding VLAN.

To enable the IGMP snooping querier feature in a VLAN, follow these steps:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip igmp snooping querier**	Enable the IGMP snooping querier.
Step 3	**ip igmp snooping querier** *ip_address*	(Optional) Specify an IP address for the IGMP snooping querier. If you do not specify an IP address, the querier tries to use the global IP address configured for the IGMP querier. **Note** The IGMP snooping querier does not generate an IGMP general query if it cannot find an IP address on the switch.
Step 4	**ip igmp snooping querier query-interval** *interval-count*	(Optional) Set the interval between IGMP queriers. The range is 1 to 18000 seconds.
Step 5	**ip igmp snooping querier tcn query** [**count** *count* \| **interval** *interval*]	(Optional) Set the time between Topology Change Notification (TCN) queries. The count range is 1 to 10. The interval range is 1 to 255 seconds.
Step 6	**ip igmp snooping querier timer expiry** *timeout*	(Optional) Set the length of time until the IGMP querier expires.The range is 60 to 300 seconds.
Step 7	**ip igmp snooping querier version** *version*	(Optional) Select the IGMP version number that the querier feature uses. Select 1 or 2.
Step 8	**end**	Return to privileged EXEC mode.
Step 9	**show ip igmp snooping vlan** *vlan-id*	(Optional) Verify that the IGMP snooping querier is enabled on the VLAN interface. The VLAN ID range is 1 to 1001 and 1006 to 4094.
Step 10	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

This example shows how to set the IGMP snooping querier source address to 10.0.0.64:

```
Switch# configure terminal
Switch(config)# ip igmp snooping querier 10.0.0.64
Switch(config)# end
```

This example shows how to set the IGMP snooping querier maximum response time to 25 seconds:

```
Switch# configure terminal
Switch(config)# ip igmp snooping querier query-interval 25
Switch(config)# end
```

This example shows how to set the IGMP snooping querier timeout to 60 seconds:

```
Switch# configure terminal
Switch(config)# ip igmp snooping querier timeout expiry 60
Switch(config)# end
```

This example shows how to set the IGMP snooping querier feature to version 2:

```
Switch# configure terminal
Switch(config)# no ip igmp snooping querier version 2
Switch(config)# end
```

Disabling IGMP Report Suppression

Note IGMP report suppression is supported only when the multicast query has IGMPv1 and IGMPv2 reports. This feature is not supported when the query includes IGMPv3 reports.

IGMP report suppression is enabled by default. When it is enabled, the switch forwards only one IGMP report per multicast router query. When report suppression is disabled, all IGMP reports are forwarded to the multicast routers.

Beginning in privileged EXEC mode, follow these steps to disable IGMP report suppression:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**no ip igmp snooping report-suppression**	Disable IGMP report suppression.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show ip igmp snooping**	Verify that IGMP report suppression is disabled.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To re-enable IGMP report suppression, use the **ip igmp snooping report-suppression** global configuration command.

Displaying IGMP Snooping Information

You can display IGMP snooping information for dynamically learned and statically configured router ports and VLAN interfaces. You can also display MAC address multicast entries for a VLAN configured for IGMP snooping.

To display IGMP snooping information, use one or more of the privileged EXEC commands in Table 20-4.

Table 20-4 Commands for Displaying IGMP Snooping Information

Command	Purpose
show ip igmp snooping [**vlan** *vlan-id*]	Display the snooping configuration information for all VLANs on the switch or for a specified VLAN. (Optional) Enter **vlan** *vlan-id* to display information for a single VLAN. The VLAN ID range is 1 to 1001 and 1006 to 4094.
show ip igmp snooping groups [**count** \|**dynamic** [**count**] \| **user** [**count**]]	Display multicast table information for the switch or about a specific parameter: • **count**—Display the total number of entries for the specified command options instead of the actual entries. • **dynamic**—Display entries learned through IGMP snooping. • **user**—Display only the user-configured multicast entries.
show ip igmp snooping groups vlan *vlan-id* [*ip_address* \| **count** \| **dynamic** [**count**] \| **user**[**count**]]	Display multicast table information for a multicast VLAN or about a specific parameter for the VLAN: • *vlan-id*—The VLAN ID range is 1 to 1001 and 1006 to 4094. • **count**—Display the total number of entries for the specified command options instead of the actual entries. • **dynamic**—Display entries learned through IGMP snooping. • *ip_address*—Display characteristics of the multicast group with the specified group IP address. • **user**—Display only the user-configured multicast entries.
show ip igmp snooping mrouter [**vlan** *vlan-id*]	Display information on dynamically learned and manually configured multicast router interfaces. **Note** When you enable IGMP snooping, the switch automatically learns the interface to which a multicast router is connected. These are dynamically learned interfaces. (Optional) Enter **vlan** *vlan-id* to display information for a single VLAN.
show ip igmp snooping querier [**vlan** *vlan-id*]	Display information about the IP address and incoming port for the most-recently received IGMP query messages in the VLAN. (Optional) Enter **vlan** *vlan-id* to display information for a single VLAN.
show ip igmp snooping querier [**vlan** *vlan-id*] **detail**	Display information about the IP address and incoming port of the most-recently received IGMP query message in the VLAN,and the configuration and operational state of the IGMP snooping querier in the VLAN.

For more information about the keywords and options in these commands, see the command reference for this release.

Understanding Multicast VLAN Registration

Multicast VLAN Registration (MVR) is designed for applications using wide-scale deployment of multicast traffic across an Ethernet ring-based service-provider network (for example, the broadcast of multiple television channels over a service-provider network). MVR allows a subscriber on a port to subscribe and unsubscribe to a multicast stream on the network-wide multicast VLAN. It allows the single multicast VLAN to be shared in the network while subscribers remain in separate VLANs. MVR provides the ability to continuously send multicast streams in the multicast VLAN, but to isolate the streams from the subscriber VLANs for bandwidth and security reasons.

MVR assumes that subscriber ports subscribe and unsubscribe (join and leave) these multicast streams by sending out IGMP join and leave messages. These messages can originate from an IGMP Version-2-compatible host with an Ethernet connection. Although MVR operates on the underlying mechanism of IGMP snooping, the two features operate independently of each other. One can be enabled or disabled without affecting the behavior of the other feature. However, if IGMP snooping and MVR are both enabled, MVR reacts only to join and leave messages from multicast groups configured under MVR. Join and leave messages from all other multicast groups are managed by IGMP snooping.

The switch CPU identifies the MVR IP multicast streams and their associated IP multicast group in the switch forwarding table, intercepts the IGMP messages, and modifies the forwarding table to include or remove the subscriber as a receiver of the multicast stream, even though the receivers might be in a different VLAN from the source. This forwarding behavior selectively allows traffic to cross between different VLANs.

You can set the switch for compatible or dynamic mode of MVR operation:

- In compatible mode, multicast data received by MVR hosts is forwarded to all MVR data ports, regardless of MVR host membership on those ports. The multicast data is forwarded only to those receiver ports that MVR hosts have joined, either by IGMP reports or by MVR static configuration. IGMP reports received from MVR hosts are never forwarded from MVR data ports that were configured in the switch.
- In dynamic mode, multicast data received by MVR hosts on the switch is forwarded from only those MVR data and client ports that the MVR hosts have joined, either by IGMP reports or by MVR static configuration. Any IGMP reports received from MVR hosts are also forwarded from all the MVR data ports in the switch. This eliminates using unnecessary bandwidth on MVR data port links, which occurs when the switch runs in compatible mode.

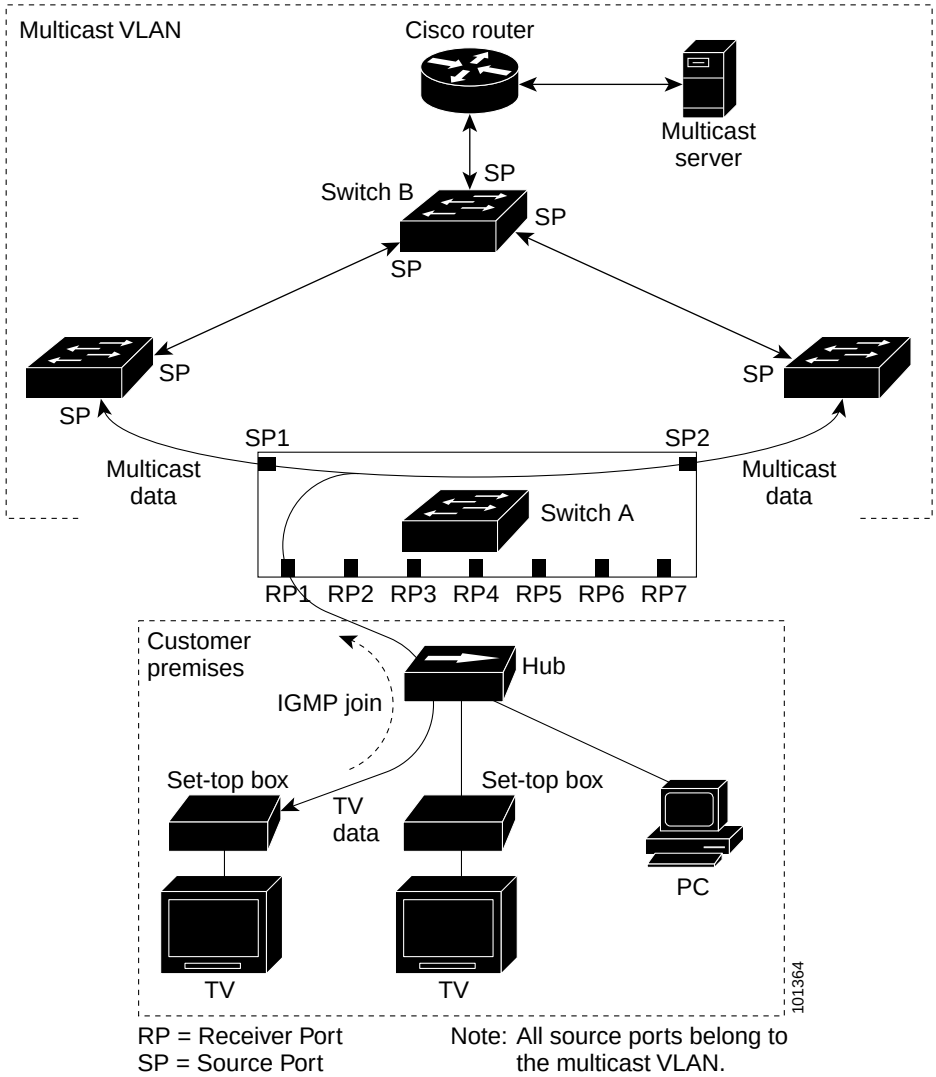
Only Layer 2 ports take part in MVR. You must configure ports as MVR receiver ports. Only one MVR multicast VLAN per switch is supported.

Using MVR in a Multicast Television Application

In a multicast television application, a PC or a television with a set-top box can receive the multicast stream. Multiple set-top boxes or PCs can be connected to one subscriber port, which is a switch port configured as an MVR receiver port. Figure 20-3 is an example configuration. DHCP assigns an IP address to the set-top box or the PC. When a subscriber selects a channel, the set-top box or PC sends an IGMP report to Switch A to join the appropriate multicast. If the IGMP report matches one of the configured IP multicast group addresses, the switch CPU modifies the hardware address table to include

this receiver port and VLAN as a forwarding destination of the specified multicast stream when it is received from the multicast VLAN. Uplink ports that send and receive multicast data to and from the multicast VLAN are called MVR source ports.

Figure 20-3 Multicast VLAN Registration Example



When a subscriber changes channels or turns off the television, the set-top box sends an IGMP leave message for the multicast stream. The switch CPU sends a MAC-based general query through the receiver port VLAN. If there is another set-top box in the VLAN still subscribing to this group, that set-top box must respond within the maximum response time specified in the query. If the CPU does not receive a response, it eliminates the receiver port as a forwarding destination for this group.

Without Immediate Leave, when the switch receives an IGMP leave message from a subscriber on a receiver port, it sends out an IGMP query on that port and waits for IGMP group membership reports. If no reports are received in a configured time period, the receiver port is removed from multicast group membership. With Immediate Leave, an IGMP query is not sent from the receiver port on which the

IGMP leave was received. As soon as the leave message is received, the receiver port is removed from multicast group membership, which speeds up leave latency. Enable the Immediate-Leave feature only on receiver ports to which a single receiver device is connected.

MVR eliminates the need to duplicate television-channel multicast traffic for subscribers in each VLAN. Multicast traffic for all channels is only sent around the VLAN trunk once—only on the multicast VLAN. The IGMP leave and join messages are in the VLAN to which the subscriber port is assigned. These messages dynamically register for streams of multicast traffic in the multicast VLAN on the Layer 3 device. Switch B. The access layer switch, Switch A, modifies the forwarding behavior to allow the traffic to be forwarded from the multicast VLAN to the subscriber port in a different VLAN, selectively allowing traffic to cross between two VLANs.

IGMP reports are sent to the same IP multicast group address as the multicast data. The Switch A CPU must capture all IGMP join and leave messages from receiver ports and forward them to the multicast VLAN of the source (uplink) port, based on the MVR mode.

Configuring MVR

These sections contain this configuration information:

- Default MVR Configuration, page 20-18
- MVR Configuration Guidelines and Limitations, page 20-19
- Configuring MVR Global Parameters, page 20-19
- Configuring MVR Interfaces, page 20-20

Default MVR Configuration

Table 20-5 shows the default MVR configuration.

Table 20-5 Default MVR Configuration

Feature	Default Setting
MVR	Disabled globally and per interface
Multicast addresses	None configured
Query response time	0.5 second
Multicast VLAN	VLAN 1
Mode	Compatible
Interface (per port) default	Neither a receiver nor a source port
Immediate Leave	Disabled on all ports

MVR Configuration Guidelines and Limitations

Follow these guidelines when configuring MVR:

- Receiver ports can only be access ports; they cannot be trunk ports. Receiver ports on a switch can be in different VLANs, but should not belong to the multicast VLAN.
- The maximum number of multicast entries (MVR group addresses) that can be configured on a switch (that is, the maximum number of television channels that can be received) is 256.
- MVR multicast data received in the source VLAN and leaving from receiver ports has its time-to-live (TTL) decremented by 1 in the switch.
- Because MVR on the switch uses IP multicast addresses instead of MAC multicast addresses, aliased IP multicast addresses are allowed on the switch. However, if the switch is interoperating with Catalyst 3550 or Catalyst 3500 XL switches, you should not configure IP addresses that alias between themselves or with the reserved IP multicast addresses (in the range 224.0.0.xxx).
- Do not configure MVR on private VLAN ports.
- MVR is not supported when multicast routing is enabled on a switch. If you enable multicast routing and a multicast routing protocol while MVR is enabled, MVR is disabled, and you receive a warning message. If you try to enable MVR while multicast routing and a multicast routing protocol are enabled, the operation to enable MVR is cancelled, and you receive an error message.
- MVR can coexist with IGMP snooping on a switch.
- MVR data received on an MVR receiver port is not forwarded to MVR source ports.
- MVR does not support IGMPv3 messages.

Configuring MVR Global Parameters

You do not need to set the optional MVR parameters if you choose to use the default settings. If you do want to change the default parameters (except for the MVR VLAN), you must first enable MVR.

Note For complete syntax and usage information for the commands used in this section, see the command reference for this release.

Beginning in privileged EXEC mode, follow these steps to configure MVR parameters:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mvr**	Enable MVR on the switch.
Step 3	**mvr group** *ip-address* [*count*]	Configure an IP multicast address on the switch or use the *count* parameter to configure a contiguous series of MVR group addresses (the range for *count* is 1 to 256; the default is 1). Any multicast data sent to this address is sent to all source ports on the switch and all receiver ports that have elected to receive data on that multicast address. Each multicast address would correspond to one television channel.

	Command	Purpose
Step 4	**mvr querytime** *value*	(Optional) Define the maximum time to wait for IGMP report memberships on a receiver port before removing the port from multicast group membership. The value is in units of tenths of a second. The range is 1 to 100, and the default is 5 tenths or one-half second.
Step 5	**mvr vlan** *vlan-id*	(Optional) Specify the VLAN in which multicast data is received; all source ports must belong to this VLAN. The VLAN range is 1 to 1001 and 1006 to 4094. The default is VLAN 1.
Step 6	**mvr mode** {**dynamic** \| **compatible**}	(Optional) Specify the MVR mode of operation: • **dynamic**—Allows dynamic MVR membership on source ports. • **compatible**—Is compatible with Catalyst 3500 XL and Catalyst 2900 XL switches and does not support IGMP dynamic joins on source ports. The default is **compatible** mode.
Step 7	**end**	Return to privileged EXEC mode.
Step 8	**show mvr** or **show mvr members**	Verify the configuration.
Step 9	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the switch to its default settings, use the **no mvr** [**mode** | **group** *ip-address* | **querytime** | **vlan**] global configuration commands.

This example shows how to enable MVR, configure the group address, set the query time to 1 second (10 tenths), specify the MVR multicast VLAN as VLAN 22, and set the MVR mode as dynamic:

```
Switch(config)# mvr
Switch(config)# mvr group 228.1.23.4
Switch(config)# mvr querytime 10
Switch(config)# mvr vlan 22
Switch(config)# mvr mode dynamic
Switch(config)# end
```

You can use the **show mvr members** privileged EXEC command to verify the MVR multicast group addresses on the switch.

Configuring MVR Interfaces

Beginning in privileged EXEC mode, follow these steps to configure Layer 2 MVR interfaces:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mvr**	Enable MVR on the switch.
Step 3	**interface** *interface-id*	Specify the Layer 2 port to configure, and enter interface configuration mode.
Step 4	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled, and NNIs are enabled.

	Command	Purpose
Step 5	**mvr type** {**source** \| **receiver**}	Configure an MVR port as one of these: • **source**—Configure uplink ports that receive and send multicast data as source ports. Subscribers cannot be directly connected to source ports. All source ports on a switch belong to the single multicast VLAN. • **receiver**—Configure a port as a receiver port if it is a subscriber port and should only receive multicast data. It does not receive data unless it becomes a member of the multicast group, either statically or by using IGMP leave and join messages. Receiver ports cannot belong to the multicast VLAN. The default configuration is as a non-MVR port. If you attempt to configure a non-MVR port with MVR characteristics, the operation fails.
Step 6	**mvr vlan** *vlan-id* **group** [*ip-address*]	(Optional) Statically configure a port to receive multicast traffic sent to the multicast VLAN and the IP multicast address. A port statically configured as a member of a group remains a member of the group until statically removed. **Note** In compatible mode, this command applies to only receiver ports. In dynamic mode, it applies to receiver ports and source ports. Receiver ports can also dynamically join multicast groups by using IGMP join and leave messages.
Step 7	**mvr immediate**	(Optional) Enable the Immediate-Leave feature of MVR on the port. **Note** This command applies to only receiver ports and should only be enabled on receiver ports to which a single receiver device is connected.
Step 8	**end**	Return to privileged EXEC mode.
Step 9	**show mvr** **show mvr interface** or **show mvr members**	Verify the configuration.
Step 10	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the interface to its default settings, use the **no mvr** [**type** | **immediate** | **vlan** *vlan-id* | **group**] interface configuration commands.

This example shows how to configure a port as a receiver port, statically configure the port to receive multicast traffic sent to the multicast group address, configure Immediate Leave on the port, and verify the results.

```
Switch(config)# mvr
Switch(config)# interface gigabitethernet0/2
Switch(config-if)# mvr type receiver
Switch(config-if)# mvr vlan 22 group 228.1.23.4
Switch(config-if)# mvr immediate
Switch(config)# end
Switch# show mvr interface
Port    Type        Status          Immediate Leave
----    ----        -------         ---------------
Gi0/2   RECEIVER    ACTIVE/DOWN     ENABLED
```

Displaying MVR Information

You can display MVR information for the switch or for a specified interface. Beginning in privileged EXEC mode, use the commands in Table 20-6 to display MVR configuration:

Table 20-6 Commands for Displaying MVR Information

Command	Purpose
show mvr	Displays MVR status and values for the switch—whether MVR is enabled or disabled, the multicast VLAN, the maximum (256) and current (0 through 256) number of multicast groups, the query response time, and the MVR mode.
show mvr interface [*interface-id*] [**members** [**vlan** *vlan-id*]]	Displays all MVR interfaces and their MVR configurations. When a specific interface is entered, displays this information: • Type—Receiver or Source • Status—One of these: – Active means the port is part of a VLAN. – Up/Down means that the port is forwarding or nonforwarding. – Inactive means that the port is not part of any VLAN. • Immediate Leave—Enabled or Disabled If the **members** keyword is entered, displays all multicast group members on this port or, if a VLAN identification is entered, all multicast group members on the VLAN. The VLAN ID range is 1 to 1001 and 1006 to 4094.
show mvr members [*ip-address*]	Displays all receiver and source ports that are members of any IP multicast group or the specified IP multicast group IP address.

Configuring IGMP Filtering and Throttling

In some environments, for example, metropolitan or multiple-dwelling unit (MDU) installations, you might want to control the set of multicast groups to which a user on a switch port can belong. You can control the distribution of multicast services, such as IP/TV, based on some type of subscription or service plan. You might also want to limit the number of multicast groups to which a user on a switch port can belong.

With the IGMP filtering feature, you can filter multicast joins on a per-port basis by configuring IP multicast profiles and associating them with individual switch ports. An IGMP profile can contain one or more multicast groups and specifies whether access to the group is permitted or denied. If an IGMP profile denying access to a multicast group is applied to a switch port, the IGMP join report requesting the stream of IP multicast traffic is dropped, and the port is not allowed to receive IP multicast traffic from that group. If the filtering action permits access to the multicast group, the IGMP report from the port is forwarded for normal processing. You can also set the maximum number of IGMP groups that a Layer 2 interface can join.

IGMP filtering controls only group-specific query and membership reports, including join and leave reports. It does not control general IGMP queries. IGMP filtering has no relationship with the function that directs the forwarding of IP multicast traffic. The filtering feature operates in the same manner whether IGMP or MVR is used to forward the multicast traffic.

IGMP filtering is applicable only to the dynamic learning of IP multicast group addresses, not static configuration.

With the IGMP throttling feature, you can set the maximum number of IGMP groups that a Layer 2 interface can join. If the maximum number of IGMP groups is set, the IGMP snooping forwarding table contains the maximum number of entries, and the interface receives an IGMP join report, you can configure an interface to drop the IGMP report or to replace the randomly selected multicast entry with the received IGMP report.

Note IGMPv3 join and leave messages are not supported on switches running IGMP filtering.

These sections contain this configuration information:

- Default IGMP Filtering and Throttling Configuration, page 20-23
- Configuring IGMP Profiles, page 20-23 (optional)
- Applying IGMP Profiles, page 20-25 (optional)
- Setting the Maximum Number of IGMP Groups, page 20-25 (optional)
- Configuring the IGMP Throttling Action, page 20-26 (optional)

Default IGMP Filtering and Throttling Configuration

Table 20-7 shows the default IGMP filtering configuration.

Table 20-7 Default IGMP Filtering Configuration

Feature	Default Setting
IGMP filters	None applied
IGMP maximum number of IGMP groups	No maximum set
IGMP profiles	None defined
IGMP profile action	Deny the range addresses

When the maximum number of groups is in forwarding table, the default IGMP throttling action is to deny the IGMP report. For configuration guidelines, see the "Configuring the IGMP Throttling Action" section on page 20-26.

Configuring IGMP Profiles

To configure an IGMP profile, use the **ip igmp profile** global configuration command with a profile number to create an IGMP profile and to enter IGMP profile configuration mode. From this mode, you can specify the parameters of the IGMP profile to be used for filtering IGMP join requests from a port. When you are in IGMP profile configuration mode, you can create the profile by using these commands:

- **deny**: Specifies that matching addresses are denied; this is the default.
- **exit**: Exits from igmp-profile configuration mode.
- **no**: Negates a command or returns to its defaults.

- **permit**: Specifies that matching addresses are permitted.
- **range**: Specifies a range of IP addresses for the profile. You can enter a single IP address or a range with a start and an end address.

The default is for the switch to have no IGMP profiles configured. When a profile is configured, if neither the **permit** nor **deny** keyword is included, the default is to deny access to the range of IP addresses.

Beginning in privileged EXEC mode, follow these steps to create an IGMP profile:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip igmp profile** *profile number*	Assign a number to the profile you are configuring, and enter IGMP profile configuration mode. The profile umber range is 1 to 4294967295.
Step 3	**permit** \| **deny**	(Optional) Set the action to permit or deny access to the IP multicast address. If no action is configured, the default for the profile is to deny access.
Step 4	**range** *ip multicast address*	Enter the IP multicast address or range of IP multicast addresses to which access is being controlled. If entering a range, enter the low IP multicast address, a space, and the high IP multicast address. You can use the **range** command multiple times to enter multiple addresses or ranges of addresses.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show ip igmp profile** *profile number*	Verify the profile configuration.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To delete a profile, use the **no ip igmp profile** *profile number* global configuration command.

To delete an IP multicast address or range of IP multicast addresses, use the **no range** *ip multicast address* IGMP profile configuration command.

This example shows how to create IGMP profile 4 allowing access to the single IP multicast address and how to verify the configuration. If the action was to deny (the default), it would not appear in the **show ip igmp profile** output display.

```
Switch(config)# ip igmp profile 4
Switch(config-igmp-profile)# permit
Switch(config-igmp-profile)# range 229.9.9.0
Switch(config-igmp-profile)# end
Switch# show ip igmp profile 4
IGMP Profile 4
    permit
    range 229.9.9.0 229.9.9.0
```

Applying IGMP Profiles

To control access as defined in an IGMP profile, use the **ip igmp filter** interface configuration command to apply the profile to the appropriate interfaces. You can apply IGMP profiles only to Layer 2 access ports; you cannot apply IGMP profiles to routed ports or SVIs. You cannot apply profiles to ports that belong to an EtherChannel port group. You can apply a profile to multiple interfaces, but each interface can have only one profile applied to it.

Beginning in privileged EXEC mode, follow these steps to apply an IGMP profile to a switch port:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the physical interface, and enter interface configuration mode. The interface must be a Layer 2 port that does not belong to an EtherChannel port group.
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled, and NNIs are enabled.
Step 4	**ip igmp filter** *profile number*	Apply the specified IGMP profile to the interface. The range is 1 to 4294967295.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config interface** *interface-id*	Verify the configuration.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove a profile from an interface, use the **no ip igmp filter** *profile number* interface configuration command.

This example shows how to apply IGMP profile 4 to a port:

```
Switch(config)# interface gigabitethernet0/2
Switch(config-if)# ip igmp filter 4
Switch(config-if)# end
```

Setting the Maximum Number of IGMP Groups

You can set the maximum number of IGMP groups that a Layer 2 interface can join by using the **ip igmp max-groups** interface configuration command. Use the **no** form of this command to set the maximum back to the default, which is no limit.

This restriction can be applied to Layer 2 ports only; you cannot set a maximum number of IGMP groups on routed ports or SVIs. You also can use this command on a logical EtherChannel interface but cannot use it on ports that belong to an EtherChannel port group.

Beginning in privileged EXEC mode, follow these steps to set the maximum number of IGMP groups in the forwarding table:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode. The interface can be a Layer 2 port that does not belong to an EtherChannel group or a EtherChannel interface.
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled, and NNIs are enabled.
Step 4	**ip igmp max-groups** *number*	Set the maximum number of IGMP groups that the interface can join. The range is 0 to 4294967294. The default is to have no maximum set.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config interface** *interface-id*	Verify the configuration.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the maximum group limitation and return to the default of no maximum, use the **no ip igmp max-groups** interface configuration command.

This example shows how to limit to 25 the number of IGMP groups that a port can join.

```
Switch(config)# interface gigabitethernet0/2
Switch(config-if)# ip igmp max-groups 25
Switch(config-if)# end
```

Configuring the IGMP Throttling Action

After you set the maximum number of IGMP groups that a Layer 2 interface can join, you can configure an interface to replace the existing group with the new group for which the IGMP report was received by using the **ip igmp max-groups action replace** interface configuration command. Use the **no** form of this command to return to the default, which is to drop the IGMP join report.

Follow these guidelines when configuring the IGMP throttling action:

- This restriction can be applied only to Layer 2 ports. You can use this command on a logical EtherChannel interface but cannot use it on ports that belong to an EtherChannel port group.
- When the maximum group limitation is set to the default (no maximum), entering the **ip igmp max-groups action** {**deny** | **replace**} command has no effect.
- If you configure the throttling action and set the maximum group limitation after an interface has added multicast entries to the forwarding table, the forwarding-table entries are either aged out or removed, depending on the throttling action.
 - If you configure the throttling action as **deny**, the entries that were previously in the forwarding table are not removed but are aged out. After these entries are aged out and the maximum number of entries is in the forwarding table, the switch drops the next IGMP report received on the interface.
 - If you configure the throttling action as **replace**, the entries that were previously in the forwarding table are removed. When the maximum number of entries is in the forwarding table, the switch replaces a randomly selected entry with the received IGMP report.

To prevent the switch from removing the forwarding-table entries, you can configure the IGMP throttling action before an interface adds entries to the forwarding table.

Beginning in privileged EXEC mode, follow these steps to configure the throttling action when the maximum number of entries is in the forwarding table:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the physical interface to be configured, and enter interface configuration mode. The interface can be a Layer 2 port that does not belong to an EtherChannel group or an EtherChannel interface. The interface cannot be a trunk port.
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled, and NNIs are enabled.
Step 4	**ip igmp max-groups action** {**deny** \| **replace**}	When an interface receives an IGMP report and the maximum number of entries is in the forwarding table, specify the action that the interface takes: • **deny**—Drop the report. • **replace**—Replace the existing group with the new group for which the IGMP report was received.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config interface** *interface-id*	Verify the configuration.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default action of dropping the report, use the **no ip igmp max-groups action** interface configuration command.

This example shows how to configure a port to remove a randomly selected multicast entry in the forwarding table and to add an IGMP group to the forwarding table when the maximum number of entries is in the table.

```
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ip igmp max-groups action replace
Switch(config-if)# end
```

Displaying IGMP Filtering and Throttling Configuration

You can display IGMP profile characteristics, and you can display the IGMP profile and maximum group configuration for all interfaces on the switch or for a specified interface. You can also display the IGMP throttling configuration for all interfaces on the switch or for a specified interface.

Use the privileged EXEC commands in Table 20-8 to display IGMP filtering and throttling configuration:

Table 20-8 Commands for Displaying IGMP Filtering and Throttling Configuration

Command	Purpose
show ip igmp profile [*profile number*]	Displays the specified IGMP profile or all the IGMP profiles defined on the switch.
show running-config [**interface** *interface-id*]	Displays the configuration of the specified interface or the configuration of all interfaces on the switch, including (if configured) the maximum number of IGMP groups to which an interface can belong and the IGMP profile applied to the interface.



Configuring Port-Based Traffic Control

This chapter describes how to configure the port-based traffic control features on the Cisco ME 3400 Ethernet Access switch.

Note For complete syntax and usage information for the commands used in this chapter, see the command reference for this release.

This chapter consists of these sections:

- Configuring Storm Control, page 21-1
- Configuring Protected Ports, page 21-5
- Configuring Port Blocking, page 21-7
- Configuring Port Security, page 21-8
- Displaying Port-Based Traffic Control Settings, page 21-17

Configuring Storm Control

These sections contain this conceptual and configuration information:

- Understanding Storm Control, page 21-1
- Default Storm Control Configuration, page 21-3
- Configuring Storm Control and Threshold Levels, page 21-3

Understanding Storm Control

Storm control prevents traffic on a LAN from being disrupted by a broadcast, multicast, or unicast storm on one of the physical interfaces. A LAN storm occurs when packets flood the LAN, creating excessive traffic and degrading network performance. Errors in the protocol-stack implementation or in the network configuration can cause a storm.

Storm control (or traffic suppression) monitors packets passing from an interface to the switching bus and determines if the packet is unicast, multicast, or broadcast. The switch counts the number of packets of a specified type received within the 1-second time interval and compares the measurement with a predefined suppression-level threshold.

Storm control uses one of these methods to measure traffic activity:

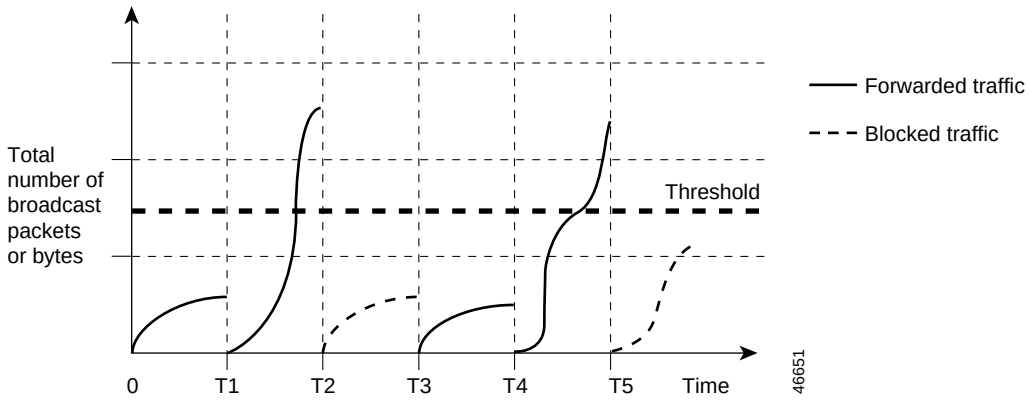
- Bandwidth as a percentage of the total available bandwidth of the port that can be used by the broadcast, multicast, or unicast traffic
- Traffic rate in packets per second at which broadcast, multicast, or unicast packets are received
- Traffic rate in bits per second at which broadcast, multicast, or unicast packets are received

With each method, the port blocks traffic when the rising threshold is reached. The port remains blocked until the traffic rate drops below the falling threshold (if one is specified) and then resumes normal forwarding. If the falling suppression level is not specified, the switch blocks all traffic until the traffic rate drops below the rising suppression level. In general, the higher the level, the less effective the protection against broadcast storms.

Note When the storm control threshold for multicast traffic is reached, all multicast traffic except control traffic, such as bridge protocol data unit (BDPU) and Cisco Discovery Protocol (CDP) frames, are blocked. However, the switch does not differentiate between routing updates, such as OSPF, and regular multicast data traffic, so both types of traffic are blocked.

The graph in Figure 21-1 shows broadcast traffic patterns on an interface over a given period of time. The example can also be applied to multicast and unicast traffic. In this example, the broadcast traffic being forwarded exceeded the configured threshold between time intervals T1 and T2 and between T4 and T5. When the amount of specified traffic exceeds the threshold, all traffic of that kind is dropped for the next time period. Therefore, broadcast traffic is blocked during the intervals following T2 and T5. At the next time interval (for example, T3), if broadcast traffic does not exceed the threshold, it is again forwarded.

Figure 21-1 Broadcast Storm Control Example



The combination of the storm-control suppression level and the 1-second time interval controls the way the storm control algorithm works. A higher threshold allows more packets to pass through. A threshold value of 100 percent means that no limit is placed on the traffic. A value of 0.0 means that all broadcast, multicast, or unicast traffic on that port is blocked.

Note Because packets do not arrive at uniform intervals, the 1-second time interval during which traffic activity is measured can affect the behavior of storm control.

You use the **storm-control** interface configuration commands to set the threshold value for each traffic type.

Default Storm Control Configuration

By default, unicast, broadcast, and multicast storm control are disabled on the switch interfaces; that is, the suppression level is 100 percent.

Configuring Storm Control and Threshold Levels

You configure storm control on a port and enter the threshold level that you want to be used for a particular type of traffic.

However, because of hardware limitations and the way in which packets of different sizes are counted, threshold percentages are approximations. Depending on the sizes of the packets making up the incoming traffic, the actual enforced threshold might differ from the configured level by several percentage points.



Note Storm control is supported only on physical interfaces. It is not supported on EtherChannel port-channels or physical interfaces that are members of port channels even though the command is available in the CLI. If a physical interface with storm control configured joins an EtherChannel, the storm control configuration for the physical interface is removed from the running configuration.

Beginning in privileged EXEC mode, follow these steps to storm control and threshold levels:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
Step 3	**no shutdown**	Enable the port, if necessary. By default, user network interfaces (UNIs) are disabled, and network node interfaces (NNIs) are enabled.

	Command	Purpose
Step 4	**storm-control** {**broadcast** \| **multicast** \| **unicast**} **level** {*level* [*level-low*] \| **bps** *bps* [*bps-low*] \| **pps** *pps* [*pps-low*]}	Configure broadcast, multicast, or unicast storm control. By default, storm control is disabled. The keywords have these meanings: • For *level*, specify the rising threshold level for broadcast, multicast, or unicast traffic as a percentage (up to two decimal places) of the bandwidth. The port blocks traffic when the rising threshold is reached. The range is 0.00 to 100.00. • (Optional) For *level-low*, specify the falling threshold level as a percentage (up to two decimal places) of the bandwidth. This value must be less than or equal to the rising suppression value. The port forwards traffic when traffic drops below this level. If you do not configure a falling suppression level, it is set to the rising suppression level. The range is 0.00 to 100.00. If you set the threshold to the maximum value (100 percent), no limit is placed on the traffic. If you set the threshold to 0.0, all broadcast, multicast, and unicast traffic on that port is blocked. • For **bps** *bps*, specify the rising threshold level for broadcast, multicast, or unicast traffic in bits per second (up to one decimal place). The port blocks traffic when the rising threshold is reached. The range is 0.0 to 10000000000.0. • (Optional) For *bps-low*, specify the falling threshold level in bits per second (up to one decimal place). It can be less than or equal to the rising threshold level. The port forwards traffic when traffic drops below this level. The range is 0.0 to 10000000000.0. • For **pps** *pps*, specify the rising threshold level for broadcast, multicast, or unicast traffic in packets per second (up to one decimal place). The port blocks traffic when the rising threshold is reached. The range is 0.0 to 10000000000.0. • (Optional) For *pps-low*, specify the falling threshold level in packets per second (up to one decimal place). It can be less than or equal to the rising threshold level. The port forwards traffic when traffic drops below this level. The range is 0.0 to 10000000000.0. For BPS and PPS settings, you can use metric suffixes such as k, m, and g for large number thresholds.
Step 5	**storm-control action** {**shutdown** \| **trap**}	Specify the action to be taken when a storm is detected. The default is to filter out the traffic and not to send traps. • Select the **shutdown** keyword to error-disable the port during a storm. • Select the **trap** keyword to generate an SNMP trap when a storm is detected.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show storm-control** [*interface-id*] [**broadcast** \| **multicast** \| **unicast**]	Verify the storm control suppression levels set on the interface for the specified traffic type. If you do not enter a traffic type, broadcast storm control settings are displayed.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable storm control, use the **no storm-control** {**broadcast** | **multicast** | **unicast**} **level** interface configuration command.

This example shows how to enable unicast storm control on a port with an 87-percent rising suppression level and a 65-percent falling suppression level:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# storm-control unicast level 87 65
```

This example shows how to enable broadcast address storm control on a port to a level of 20 percent. When the broadcast traffic exceeds the configured level of 20 percent of the total available bandwidth of the port within the traffic-storm-control interval, the switch drops all broadcast traffic until the end of the traffic-storm-control interval:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# storm-control broadcast level 20
```

Configuring Protected Ports

Some applications require that no traffic be forwarded at Layer 2 between ports on the same switch so that one neighbor does not see the traffic generated by another neighbor. In such an environment, the use of protected ports ensures that there is no exchange of unicast, broadcast, or multicast traffic between these ports on the switch.

Note NNI ports default to non-protected ports. Since UNI ports provide port isolation, protected port is not available on UNI ports.

More more information about UNIs, see the "UNI and NNI Ports" section on page 9-3.

Protected ports have these features:

- A protected port does not forward any traffic (unicast, multicast, or broadcast) to any other port that is also a protected port. Data traffic cannot be forwarded between protected ports at Layer 2; only control traffic, such as PIM packets, is forwarded because these packets are processed by the CPU and forwarded in software. All data traffic passing between protected ports must be forwarded through a Layer 3 device.
- Forwarding behavior between a protected port and a nonprotected port proceeds as usual.

These sections contain this configuration information:

- Default Protected Port Configuration, page 21-5
- Protected Port Configuration Guidelines, page 21-6
- Configuring a Protected Port, page 21-6

Default Protected Port Configuration

The default is to have no protected ports defined.

Protected Port Configuration Guidelines

You can configure protected ports on a physical interface that is configured as an NNI (for example, Gigabit Ethernet port 1) or an EtherChannel group (for example, port-channel 5). When you enable protected ports for a port channel, it is enabled for all ports in the port-channel group.

Do not configure a private-VLAN port as a protected port. Do not configure a protected port as a private-VLAN port. A private-VLAN isolated port does not forward traffic to other isolated ports or community ports. For more information about private VLANs, see Chapter 12, "Configuring Private VLANs."

Configuring a Protected Port

Beginning in privileged EXEC mode, follow these steps to define a port as a protected port:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to configure, and enter interface configuration mode. The interface must be an NNI. **Note** By default, UNIs are protected ports.
Step 3	**switchport protected**	Configure the interface to be a protected port.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show interfaces** *interface-id* **switchport**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable protected port, use the **no switchport protected** interface configuration command.

This example shows how to configure a port as a protected port:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# switchport protected
Switch(config-if)# end
```

This example shows how to configure a FastEthernet port as a protected port.

```
Switch# configure terminal
Switch(config)# interface fastethernet 0/1
Switch(config-if)# port-type NNI
Switch(config-if)# no shutdown
Switch(config-if)# switchport protected
Switch(config-if)# end
```

Note There can only be four NNIs on the Cisco ME switch at the same time.

Configuring Port Blocking

By default, the switch floods packets with unknown destination MAC addresses out of all ports. If unknown unicast and multicast traffic is forwarded to a protected port, there could be security issues. To prevent unknown unicast or multicast traffic from being forwarded from one port to another, you can block a port (protected or nonprotected) from flooding unknown unicast or multicast packets to other ports.

These sections contain this configuration information:

- Default Port Blocking Configuration, page 21-7
- Blocking Flooded Traffic on an Interface, page 21-7

Default Port Blocking Configuration

The default is to not block flooding of unknown multicast and unicast traffic out of a port, but to flood these packets to all ports.

Blocking Flooded Traffic on an Interface

Note The interface can be a physical interface or an EtherChannel group. When you block multicast or unicast traffic for a port channel, it is blocked on all ports in the port-channel group.

Beginning in privileged EXEC mode, follow these steps to disable the flooding of multicast and unicast packets out of an interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled, and NNIs are enabled.
Step 4	**switchport block multicast**	Block unknown multicast forwarding out of the port.
Step 5	**switchport block unicast**	Block unknown unicast forwarding out of the port.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show interfaces** *interface-id* **switchport**	Verify your entries.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the interface to the default condition where no traffic is blocked and normal forwarding occurs on the port, use the **no switchport block** {**multicast** | **unicast**} interface configuration commands.

This example shows how to block unicast and multicast flooding on a port:

```
Switch# configure terminal
Switch(config)# interface fastethernet0/1
Switch(config-if)# no shutdown
Switch(config-if)# switchport block multicast
Switch(config-if)# switchport block unicast
Switch(config-if)# end
```

Configuring Port Security

You can use the port security feature to restrict input to an interface by limiting and identifying MAC addresses of the stations allowed to access the port. When you assign secure MAC addresses to a secure port, the port does not forward packets with source addresses outside the group of defined addresses. If you limit the number of secure MAC addresses to one and assign a single secure MAC address, the workstation attached to that port is assured the full bandwidth of the port.

If a port is configured as a secure port and the maximum number of secure MAC addresses is reached, when the MAC address of a station attempting to access the port is different from any of the identified secure MAC addresses, a security violation occurs. Also, if a station with a secure MAC address configured or learned on one secure port attempts to access another secure port, a violation is flagged.

These sections contain this conceptual and configuration information:

- Understanding Port Security, page 21-8
- Default Port Security Configuration, page 21-10
- Port Security Configuration Guidelines, page 21-10
- Enabling and Configuring Port Security, page 21-11
- Enabling and Configuring Port Security Aging, page 21-15

Understanding Port Security

These sections contain this conceptual information:

- Secure MAC Addresses, page 21-8
- Security Violations, page 21-9

Secure MAC Addresses

You configure the maximum number of secure addresses allowed on a port by using the **switchport port-security maximum** *value* interface configuration command.

Note If you try to set the maximum value to a number less than the number of secure addresses already configured on an interface, the command is rejected.

The switch supports these types of secure MAC addresses:

- Static secure MAC addresses—These are manually configured by using the **switchport port-security mac-address** *mac-address* interface configuration command, stored in the address table, and added to the switch running configuration.

- Dynamic secure MAC addresses—These are dynamically configured, stored only in the address table, and removed when the switch restarts.
- *Sticky* secure MAC addresses—These can be dynamically learned or manually configured, stored in the address table, and added to the running configuration. If these addresses are saved in the configuration file, when the switch restarts, the interface does not need to dynamically reconfigure them.

You can configure an interface to convert the dynamic MAC addresses to sticky secure MAC addresses and to add them to the running configuration by enabling *sticky learning*. To enable sticky learning, enter the **switchport port-security mac-address sticky** interface configuration command. When you enter this command, the interface converts all the dynamic secure MAC addresses, including those that were dynamically learned before sticky learning was enabled, to sticky secure MAC addresses. All sticky secure MAC addresses are added to the running configuration.

The sticky secure MAC addresses do not automatically become part of the configuration file, which is the startup configuration used each time the switch restarts. If you save the sticky secure MAC addresses in the configuration file, when the switch restarts, the interface does not need to relearn these addresses. If you do not save the sticky secure addresses, they are lost.

If sticky learning is disabled, the sticky secure MAC addresses are converted to dynamic secure addresses and are removed from the running configuration.

The maximum number of secure MAC addresses that you can configure on a switch is set by the maximum number of available MAC addresses allowed in the system. This number is determined by the active Switch Database Management (SDM) template. See Chapter 6, “Configuring SDM Templates.” This number is the total of available MAC addresses, including those used for other Layer 2 functions and any other secure MAC addresses configured on interfaces.

Security Violations

It is a security violation when one of these situations occurs:

- The maximum number of secure MAC addresses have been added to the address table, and a station whose MAC address is not in the address table attempts to access the interface.
- An address learned or configured on one secure interface is seen on another secure interface in the same VLAN.

You can configure the interface for one of three violation modes, based on the action to be taken if a violation occurs:

- protect—when the number of secure MAC addresses reaches the maximum limit allowed on the port, packets with unknown source addresses are dropped until you remove a sufficient number of secure MAC addresses to drop below the maximum value or increase the number of maximum allowable addresses. You are not notified that a security violation has occurred.



Note We do not recommend configuring the protect violation mode on a trunk port. The protect mode disables learning when any VLAN reaches its maximum limit, even if the port has not reached its maximum limit.

- restrict—when the number of secure MAC addresses reaches the maximum limit allowed on the port, packets with unknown source addresses are dropped until you remove a sufficient number of secure MAC addresses to drop below the maximum value or increase the number of maximum allowable addresses. In this mode, you are notified that a security violation has occurred. An SNMP trap is sent, a syslog message is logged, and the violation counter increments.

- shutdown—a port security violation causes the interface to become error-disabled and to shut down immediately, and the port LED turns off. An SNMP trap is sent, a syslog message is logged, and the violation counter increments. When a secure port is in the error-disabled state, you can bring it out of this state by entering the **errdisable recovery cause** *psecure-violation* global configuration command, or you can manually re-enable it by entering the **shutdown** and **no shut down** interface configuration commands. This is the default mode.

Table 21-1 shows the violation mode and the actions taken when you configure an interface for port security.

Table 21-1 Security Violation Mode Actions

Violation Mode	Traffic is forwarded[1]	Sends SNMP trap	Sends syslog message	Displays error message[2]	Violation counter increments	Shuts down port
protect	No	No	No	No	No	No
restrict	No	Yes	Yes	No	Yes	No
shutdown	No	Yes	Yes	No	Yes	Yes

1. Packets with unknown source addresses are dropped until you remove a sufficient number of secure MAC addresses.
2. The switch returns an error message if you manually configure an address that would cause a security violation.

Default Port Security Configuration

Table 21-2 shows the default port security configuration for an interface.

Table 21-2 Default Port Security Configuration

Feature	Default Setting
Port security	Disabled on a port.
Sticky address learning	Disabled.
Maximum number of secure MAC addresses per port	1.
Violation mode	Shutdown. The port shuts down when the maximum number of secure MAC addresses is exceeded.
Port security aging	Disabled. Aging time is 0. Static aging is disabled. Type is absolute.

Port Security Configuration Guidelines

Follow these guidelines when configuring port security:

- Port security can only be configured on static access ports or trunk ports. A secure port cannot be a dynamic access port.
- A secure port cannot be a destination port for Switched Port Analyzer (SPAN).
- A secure port cannot belong to a Fast EtherChannel or a Gigabit EtherChannel port group.
- A secure port cannot be a private-VLAN port.

- When you enter a maximum secure address value for an interface, and the new value is greater than the previous value, the new value overwrites the previously configured value. If the new value is less than the previous value and the number of configured secure addresses on the interface exceeds the new value, the command is rejected.
- The switch does not support port security aging of sticky secure MAC addresses.

Table 21-3 summarizes port security compatibility with other port-based features.

Table 21-3 Port Security Compatibility with Other Switch Features

Type of Port or Feature on Port	Compatible with Port Security
Trunk port	Yes
Dynamic-access port (a VLAN Query Protocol [VQP] port configured with the switchport access vlan dynamic interface configuration command)	No
Routed port	No
SPAN source port	Yes
SPAN destination port	No
EtherChannel	No
Tunneling port	Yes
Protected port	Yes
802.1x port	Yes
Private VLAN port	No
IP source guard	Yes
Dynamic Address Resolution Protocol (ARP) inspection	Yes
Flex Links	Yes

Enabling and Configuring Port Security

Beginning in privileged EXEC mode, follow these steps to restrict input to an interface by limiting and identifying MAC addresses of the stations allowed to access the port:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled, and NNIs are enabled.
Step 4	**switchport mode** {**access** \| **trunk**}	Set the interface switchport mode as access or trunk; an interface in the default mode (dynamic auto) cannot be configured as a secure port.
Step 5	**switchport port-security**	Enable port security on the interface.

	Command	Purpose
Step 6	**switchport port-security** [**maximum** *value* [**vlan** *vlan-list* \| **access**]	(Optional) Set the maximum number of secure MAC addresses for the interface. The maximum number of secure MAC addresses that you can configure on a switch is set by the maximum number of available MAC addresses allowed in the system. This number is set by the active Switch Database Management (SDM) template. See Chapter 6, "Configuring SDM Templates." This number is the total of available MAC addresses, including those used for other Layer 2 functions and any other secure MAC addresses configured on interfaces. (Optional) **vlan**—set a per-VLAN maximum value. Enter one of these options after you enter the **vlan** keyword: • *vlan-list*—On a trunk port, you can set a per-VLAN maximum value on a range of VLANs separated by a hyphen or a series of VLANs separated by commas. For nonspecified VLANs, the per-VLAN maximum value is used. • **access**—On an access port, specify the VLAN as an access VLAN.
Step 7	**switchport port-security violation** {**protect** \| **restrict** \| **shutdown**}	(Optional) Set the violation mode, the action to be taken when a security violation is detected, as one of these: • **protect**—When the number of port secure MAC addresses reaches the maximum limit allowed on the port, packets with unknown source addresses are dropped until you remove a sufficient number of secure MAC addresses to drop below the maximum value or increase the number of maximum allowable addresses. You are not notified that a security violation has occurred. **Note** We do not recommend configuring the protect mode on a trunk port. The protect mode disables learning when any VLAN reaches its maximum limit, even if the port has not reached its maximum limit. • **restrict**—When the number of secure MAC addresses reaches the limit allowed on the port, packets with unknown source addresses are dropped until you remove a sufficient number of secure MAC addresses or increase the number of maximum allowable addresses. An SNMP trap is sent, a syslog message is logged, and the violation counter increments. • **shutdown**—The interface is error-disabled when a violation occurs, and the port LED turns off. An SNMP trap is sent, a syslog message is logged, and the violation counter increments. **Note** When a secure port is in the error-disabled state, you can bring it out of this state by entering the **errdisable recovery cause psecure-violation** global configuration command, or you can manually re-enable it by entering the **shutdown** and **no shutdown** interface configuration commands.

	Command	Purpose
Step 8	**switchport port-security** [**mac-address** *mac-address* [**vlan** {*vlan-id* \| {**access**}}]	(Optional) Enter a secure MAC address for the interface. You can use this command to enter the maximum number of secure MAC addresses. If you configure fewer secure MAC addresses than the maximum, the remaining MAC addresses are dynamically learned. **Note** If you enable sticky learning after you enter this command, the secure addresses that were dynamically learned are converted to sticky secure MAC addresses and are added to the running configuration. (Optional) **vlan**—set a per-VLAN maximum value. Enter one of these options after you enter the **vlan** keyword: • *vlan-id*—On a trunk port, you can specify the VLAN ID and the MAC address. If you do not specify a VLAN ID, the native VLAN is used. • **access**—On an access port, specify the VLAN as an access VLAN.
Step 9	**switchport port-security mac-address sticky**	(Optional) Enable sticky learning on the interface.
Step 10	**switchport port-security mac-address sticky** [*mac-address* \| **vlan** {*vlan-id* \| {**access**}]	(Optional) Enter a sticky secure MAC address, repeating the command as many times as necessary. If you configure fewer secure MAC addresses than the maximum, the remaining MAC addresses are dynamically learned, are converted to sticky secure MAC addresses, and are added to the running configuration. **Note** If you do not enable sticky learning before this command is entered, an error message appears, and you cannot enter a sticky secure MAC address. (Optional) **vlan**—set a per-VLAN maximum value. Enter one of these options after you enter the **vlan** keyword: • *vlan-id*—On a trunk port, you can specify the VLAN ID and the MAC address. If you do not specify a VLAN ID, the native VLAN is used. • **access**—On an access port, specify the VLAN as an access VLAN.
Step 11	**end**	Return to privileged EXEC mode.
Step 12	**show port-security**	Verify your entries.
Step 13	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the interface to the default condition as not a secure port, use the **no switchport port-security** interface configuration command. If you enter this command when sticky learning is enabled, the sticky secure addresses remain part of the running configuration but are removed from the address table. All addresses are now dynamically learned.

To return the interface to the default number of secure MAC addresses, use the **no switchport port-security maximum** *value* interface configuration command. To return the violation mode to the default condition (shutdown mode), use the **no switchport port-security violation** {**protocol** | **restrict**} interface configuration command.

To disable sticky learning on an interface, use the **no switchport port-security mac-address sticky** interface configuration command. The interface converts the sticky secure MAC addresses to dynamic secure addresses. However, if you have previously saved the configuration with the sticky MAC addresses, you should save the configuration again after entering the **no switchport port-security mac-address sticky** command, or the sticky addresses will be restored if the switch reboots.

Use the **clear port-security** {**all** | **configured | dynamic | sticky**} privileged EXEC command to delete from the MAC address table all secure addresses or all secure addresses of a specific type (configured, dynamic, or sticky) on the switch or on an interface.

To delete a specific secure MAC address from the address table, use the **no switchport port-security mac-address** *mac-address* interface configuration command.

To delete all dynamic secure addresses on an interface from the address table, enter the **no switchport port-security** interface configuration command followed by the **switchport port-security** command (to re-enable port security on the interface). If you use the **no switchport port-security mac-address sticky** interface configuration command to convert sticky secure MAC addresses to dynamic secure MAC addresses before entering the **no switchport port-security** command, all secure addresses on the interface except those that were manually configured are deleted.

You must specifically delete configured secure MAC addresses from the address table by using the **no switchport port-security mac-address** *mac-address* interface configuration command.

This example shows how to enable port security on a port and to set the maximum number of secure addresses to 50. The violation mode is the default, no static secure MAC addresses are configured, and sticky learning is enabled.

```
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# switchport mode access
Switch(config-if)# switchport port-security
Switch(config-if)# switchport port-security maximum 50
Switch(config-if)# switchport port-security mac-address sticky
```

This example shows how to configure a static secure MAC address on VLAN 3 on a port:

```
Switch(config)# interface gigabitethernet0/2
Switch(config-if)# switchport mode trunk
Switch(config-if)# switchport port-security
Switch(config-if)# switchport port-security mac-address 0000.02000.0004 vlan 3
```

This example shows how to enable sticky port security on a port, to manually configure MAC addresses for data VLAN, and to set the total maximum number of secure addresses to 10.

```
Switch(config)# interface FastEthernet0/1
Switch(config-if)# no shutdown
Switch(config-if)# switchport access vlan 21
Switch(config-if)# switchport mode access
Switch(config-if)# switchport port-security
Switch(config-if)# switchport port-security maximum 10
Switch(config-if)# switchport port-security violation restrict
Switch(config-if)# switchport port-security mac-address sticky
Switch(config-if)# switchport port-security mac-address sticky 0000.0000.0002
Switch(config-if)# switchport port-security mac-address 0000.0000.0003
Switch(config-if)# switchport port-security maximum 10 vlan access
```

Enabling and Configuring Port Security Aging

You can use port security aging to set the aging time for all secure addresses on a port. Two types of aging are supported per port:

- Absolute—The secure addresses on the port are deleted after the specified aging time.
- Inactivity—The secure addresses on the port are deleted only if the secure addresses are inactive for the specified aging time.

Use this feature to remove and add devices on a secure port without manually deleting the existing secure MAC addresses and to still limit the number of secure addresses on a port. You can enable or disable the aging of secure addresses on a per-port basis.

Beginning in privileged EXEC mode, follow these steps to configure port security aging:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled, and NNIs are enabled.

	Command	Purpose
Step 4	**switchport port-security aging** {**static** \| **time** *time* \| **type** {**absolute** \| **inactivity**}}	Enable or disable static aging for the secure port, or set the aging time or type. **Note** The switch does not support port security aging of sticky secure addresses. Enter **static** to enable aging for statically configured secure addresses on this port. For *time*, specify the aging time for this port. The valid range is from 0 to 1440 minutes. For **type**, select one of these keywords: • **absolute**—Sets the aging type as absolute aging. All the secure addresses on this port age out exactly after the time (minutes) specified lapses and are removed from the secure address list. • **inactivity**—Sets the aging type as inactivity aging. The secure addresses on this port age out only if there is no data traffic from the secure source addresses for the specified time period.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show port-security** [**interface** *interface-id*] [**address**]	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable port security aging for all secure addresses on a port, use the **no switchport port-security aging time** interface configuration command. To disable aging for only statically configured secure addresses, use the **no switchport port-security aging static** interface configuration command.

This example shows how to set the aging time as 2 hours for the secure addresses on a port:

```
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# switchport port-security aging time 120
```

This example shows how to set the aging time as 2 minutes for the inactivity aging type with aging enabled for the configured secure addresses on the interface:

```
Switch(config-if)# switchport port-security aging time 2
Switch(config-if)# switchport port-security aging type inactivity
Switch(config-if)# switchport port-security aging static
```

You can verify the previous commands by entering the **show port-security interface** *interface-id* privileged EXEC command.

Displaying Port-Based Traffic Control Settings

The **show interfaces** *interface-id* **switchport** privileged EXEC command displays (among other characteristics) the interface traffic suppression and control configuration. The **show storm-control** and **show port-security** privileged EXEC commands display those storm control and port security settings.

To display traffic control information, use one or more of the privileged EXEC commands in Table 21-4.

Table 21-4 Commands for Displaying Traffic Control Status and Configuration

Command	Purpose
show interfaces [*interface-id*] **switchport**	Displays the administrative and operational status of all switching (nonrouting) ports or the specified port, including port blocking and port protection settings.
show storm-control [*interface-id*] [**broadcast** \| **multicast** \| **unicast**]	Displays storm control suppression levels set on all interfaces or the specified interface for the specified traffic type or for broadcast traffic if no traffic type is entered.
show port-security [**interface** *interface-id*]	Displays port security settings for the switch or for the specified interface, including the maximum allowed number of secure MAC addresses for each interface, the number of secure MAC addresses on the interface, the number of security violations that have occurred, and the violation mode.
show port-security [**interface** *interface-id*] **address**	Displays all secure MAC addresses configured on all switch interfaces or on a specified interface with aging information for each address.
show port-security interface *interface-id* **vlan**	Displays the number of secure MAC addresses configured per VLAN on the specified interface.



Configuring CDP

This chapter describes how to configure Cisco Discovery Protocol (CDP) on the Cisco ME 3400 Ethernte Access switch.

Note For complete syntax and usage information for the commands used in this chapter, see the command reference for this release and the "System Management Commands" section in the *Cisco IOS Configuration Fundamentals Command Reference, Release 12.2*.

This chapter consists of these sections:

- Understanding CDP, page 22-1
- Configuring CDP, page 22-2
- Monitoring and Maintaining CDP, page 22-5

Understanding CDP

CDP is a device discovery protocol that runs over Layer 2 (the data link layer) on all Cisco-manufactured devices (routers, bridges, access servers, and switches) and allows network management applications to discover Cisco devices that are neighbors of already known devices. With CDP, network management applications can learn the device type and the Simple Network Management Protocol (SNMP) agent address of neighboring devices running lower-layer, transparent protocols. This feature enables applications to send SNMP queries to neighboring devices.

CDP runs on all media that support Subnetwork Access Protocol (SNAP). Because CDP runs over the data-link layer only, two systems that support different network-layer protocols can learn about each other.

Note On the Cisco ME switch, CDP runs only on NNIs.

Each CDP-configured device sends periodic messages to a multicast address, advertising at least one address at which it can receive SNMP messages. The advertisements also contain time-to-live, or holdtime information, which is the length of time a receiving device holds CDP information before discarding it. Each device also listens to the messages sent by other devices to learn about neighboring devices.

The switch supports CDP Version 2.

Configuring CDP

These sections contain this configuration information:

- Default CDP Configuration, page 22-2
- Configuring the CDP Characteristics, page 22-2
- Disabling and Enabling CDP, page 22-3
- Disabling and Enabling CDP on an Interface, page 22-4

Default CDP Configuration

Table 22-1 shows the default CDP configuration.

Table 22-1 Default CDP Configuration

Feature	Default Setting
CDP global state	Enabled only on NNIs
CDP interface state	Enabled only on NNIs
CDP timer (packet update frequency)	60 seconds
CDP holdtime (before discarding)	180 seconds
CDP Version-2 advertisements	Enabled

Configuring the CDP Characteristics

You can configure the frequency of CDP updates, the amount of time to hold the information before discarding it, and whether or not to send Version-2 advertisements.

Beginning in privileged EXEC mode, follow these steps to configure the CDP timer, holdtime, and advertisement type.

Note Steps 2 through 4 are all optional and can be performed in any order.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**cdp timer** *seconds*	(Optional) Set the transmission frequency of CDP updates in seconds. The range is 5 to 254; the default is 60 seconds.
Step 3	**cdp holdtime** *seconds*	(Optional) Specify the amount of time a receiving device should hold the information sent by your device before discarding it. The range is 10 to 255 seconds; the default is 180 seconds.
Step 4	**cdp advertise-v2**	(Optional) Configure CDP to send Version-2 advertisements. This is the default state.
Step 5	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 6	**show cdp**	Verify your settings.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no** form of the CDP commands to return to the default settings.

This example shows how to configure CDP characteristics.

```
Switch# configure terminal
Switch(config)# cdp timer 50
Switch(config)# cdp holdtime 120
Switch(config)# cdp advertise-v2
Switch(config)# end
```

For additional CDP **show** commands, see the "Monitoring and Maintaining CDP" section on page 22-5.

Disabling and Enabling CDP

CDP is enabled by default on NNIs.

Note Cisco devices (such as Cisco IP Phones) regularly exchange CDP messages with connected devices. Disabling CDP can interrupt device connectivity.

Beginning in privileged EXEC mode, follow these steps to disable the CDP device discovery capability:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**no cdp run**	Disable CDP.
Step 3	**end**	Return to privileged EXEC mode.

Beginning in privileged EXEC mode, follow these steps to enable CDP when it has been disabled:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**cdp run**	Enable CDP after disabling it.
Step 3	**end**	Return to privileged EXEC mode.

This example shows how to enable CDP if it has been disabled.

```
Switch# configure terminal
Switch(config)# cdp run
Switch(config)# end
```

Disabling and Enabling CDP on an Interface

CDP is enabled by default on all supported interfaces to send and to receive CDP information.

Beginning in privileged EXEC mode, follow these steps to disable CDP on a port:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface on which you are disabling CDP, and enter interface configuration mode. **Note** If the interface is a UNI, you must enter the **port-type nni** interface configuration command before configuring CDP.
Step 3	**no cdp enable**	Disable CDP on the interface.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Beginning in privileged EXEC mode, follow these steps to enable CDP on a port when it has been disabled:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface on which you are enabling CDP, and enter interface configuration mode. **Note** If the interface is a UNI, you must enter the **port-type nni** interface configuration command before configuring CDP. CDP is enabled on NNIs by default.
Step 3	**cdp enable**	Enable CDP on the interface after disabling it.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

This example shows how to enable CDP on a port when it has been disabled.

```
Switch# configure terminal
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# cdp enable
Switch(config-if)# end
```

Monitoring and Maintaining CDP

To monitor and maintain CDP on your device, perform one or more of these tasks, beginning in privileged EXEC mode.

Command	Description
clear cdp counters	Reset the traffic counters to zero.
clear cdp table	Delete the CDP table of information about neighbors.
show cdp	Display global information, such as frequency of transmissions and the holdtime for packets being sent.
show cdp entry *entry-name* [**protocol** \| **version**]	Display information about a specific neighbor. You can enter an asterisk (*) to display all CDP neighbors, or you can enter the name of the neighbor about which you want information. You can also limit the display to information about the protocols enabled on the specified neighbor or information about the version of software running on the device.
show cdp interface [*interface-id*]	Display information about interfaces where CDP is enabled. You can limit the display to the interface about which you want information.
show cdp neighbors [*interface-id*] [**detail**]	Display information about neighbors, including device type, interface type and number, holdtime settings, capabilities, platform, and port ID. You can limit the display to neighbors of a specific interface or expand the display to provide more detailed information.
show cdp traffic	Display CDP counters, including the number of packets sent and received and checksum errors.



CHAPTER 23

Configuring UDLD

This chapter describes how to configure the UniDirectional Link Detection (UDLD) protocol on the Cisco ME 3400 Ethernet Access switch.

Note For complete syntax and usage information for the commands used in this chapter, see the command reference for this release.

This chapter consists of these sections:

- Understanding UDLD, page 23-1
- Configuring UDLD, page 23-4
- Displaying UDLD Status, page 23-6

Understanding UDLD

UDLD is a Layer 2 protocol that enables devices connected through fiber-optic or twisted-pair Ethernet cables to monitor the physical configuration of the cables and detect when a unidirectional link exists. All connected devices must support UDLD for the protocol to successfully identify and disable unidirectional links. When UDLD detects a unidirectional link, it disables the affected port and alerts you. Unidirectional links can cause a variety of problems, including spanning-tree topology loops.

Modes of Operation

UDLD supports two modes of operation: normal (the default) and aggressive. In normal mode, UDLD can detect unidirectional links due to misconnected ports on fiber-optic connections. In aggressive mode, UDLD can also detect unidirectional links due to one-way traffic on fiber-optic and twisted-pair links and to misconnected ports on fiber-optic links.

In normal and aggressive modes, UDLD works with the Layer 1 mechanisms to learn the physical status of a link. At Layer 1, autonegotiation takes care of physical signaling and fault detection. UDLD performs tasks that autonegotiation cannot perform, such as detecting the identities of neighbors and shutting down misconnected ports. When you enable both autonegotiation and UDLD, the Layer 1 and Layer 2 detections work together to prevent physical and logical unidirectional connections and the malfunctioning of other protocols.

A unidirectional link occurs whenever traffic sent by a local device is received by its neighbor but traffic from the neighbor is not received by the local device.

In normal mode, UDLD detects a unidirectional link when fiber strands in a fiber-optic port are misconnected and the Layer 1 mechanisms do not detect this misconnection. If the ports are connected correctly but the traffic is one way, UDLD does not detect the unidirectional link because the Layer 1 mechanism, which is supposed to detect this condition, does not do so. In this case, the logical link is considered undetermined, and UDLD does not disable the port.

When UDLD is in normal mode, if one of the fiber strands in a pair is disconnected and autonegotiation is active, the link does not stay up because the Layer 1 mechanisms did not detect a physical problem with the link. In this case, UDLD does not take any action, and the logical link is considered undetermined.

In aggressive mode, UDLD detects a unidirectional link by using the previous detection methods. UDLD in aggressive mode can also detect a unidirectional link on a point-to-point link on which no failure between the two devices is allowed. It can also detect a unidirectional link when one of these problems exists:

- On fiber-optic or twisted-pair links, one of the ports cannot send or receive traffic.
- On fiber-optic or twisted-pair links, one of the ports is down while the other is up.
- One of the fiber strands in the cable is disconnected.

In these cases, UDLD disables the affected port.

In a point-to-point link, UDLD hello packets can be considered as a heart beat whose presence guarantees the health of the link. Conversely, the loss of the heart beat means that the link must be shut down if it is not possible to re-establish a bidirectional link.

If both fiber strands in a cable are working normally from a Layer 1 perspective, UDLD in aggressive mode detects whether those fiber strands are connected correctly and whether traffic is flowing bidirectionally between the correct neighbors. This check cannot be performed by autonegotiation because autonegotiation operates at Layer 1.

Methods to Detect Unidirectional Links

UDLD operates by using two mechanisms:

- Neighbor database maintenance

 UDLD learns about other UDLD-capable neighbors by periodically sending a hello packet (also called an advertisement or probe) on every active port to keep each device informed about its neighbors.

 When the switch receives a hello message, it caches the information until the age time (hold time or time-to-live) expires. If the switch receives a new hello message before an older cache entry ages, the switch replaces the older entry with the new one.

 Whenever a port is disabled and UDLD is running, whenever UDLD is disabled on a port, or whenever the switch is reset, UDLD clears all existing cache entries for the ports affected by the configuration change. UDLD sends at least one message to inform the neighbors to flush the part of their caches affected by the status change. The message is intended to keep the caches synchronized.

- Event-driven detection and echoing

 UDLD relies on echoing as its detection mechanism. Whenever a UDLD device learns about a new neighbor or receives a resynchronization request from an out-of-sync neighbor, it restarts the detection window on its side of the connection and sends echo messages in reply. Because this behavior is the same on all UDLD neighbors, the sender of the echoes expects to receive an echo in reply.

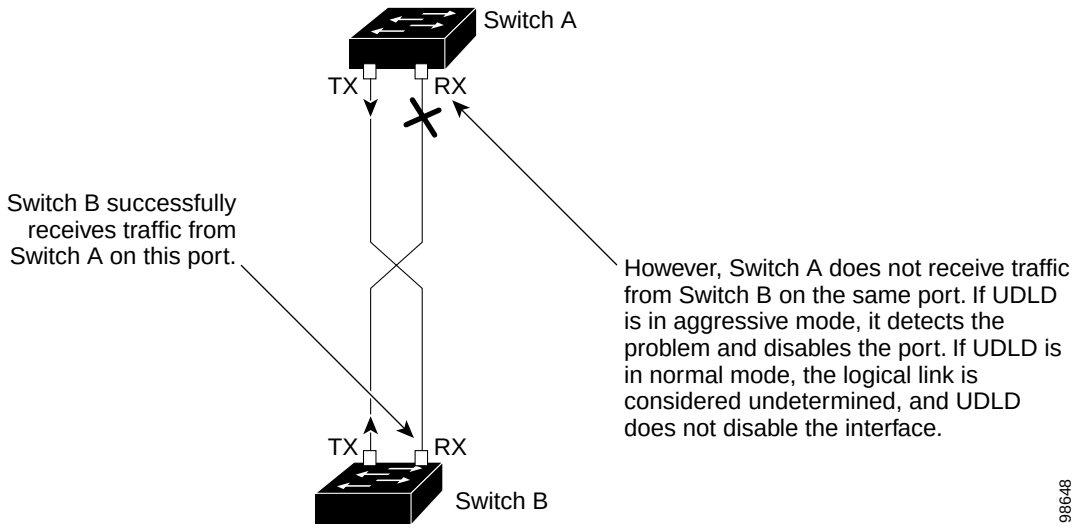
 If the detection window ends and no valid reply message is received, the link might shut down, depending on the UDLD mode. When UDLD is in normal mode, the link might be considered undetermined and might not be shut down. When UDLD is in aggressive mode, the link is considered unidirectional, and the port is disabled.

If UDLD in normal mode is in the advertisement or in the detection phase and all the neighbor cache entries are aged out, UDLD restarts the link-up sequence to resynchronize with any potentially out-of-sync neighbors.

If you enable aggressive mode when all the neighbors of a port have aged out either in the advertisement or in the detection phase, UDLD restarts the link-up sequence to resynchronize with any potentially out-of-sync neighbor. UDLD shuts down the port if, after the fast train of messages, the link state is still undetermined.

Figure 23-1 shows an example of a unidirectional link condition.

Figure 23-1 UDLD Detection of a Unidirectional Link



Configuring UDLD

These sections contain this configuration information:

- Default UDLD Configuration, page 23-4
- Configuration Guidelines, page 23-4
- Enabling UDLD Globally, page 23-5
- Enabling UDLD on an Interface, page 23-5
- Resetting an Interface Disabled by UDLD, page 23-6

Default UDLD Configuration

Table 23-1 shows the default UDLD configuration.

Table 23-1 Default UDLD Configuration

Feature	Default Setting
UDLD global enable state	Globally disabled
UDLD per-port enable state for fiber-optic media	Disabled on all Ethernet fiber-optic ports
UDLD per-port enable state for twisted-pair (copper) media	Disabled on all Ethernet 10/100 and 1000BASE-TX ports
UDLD aggressive mode	Disabled

Configuration Guidelines

These are the UDLD configuration guidelines:

- UDLD is not supported on ATM ports.
- A UDLD-capable port cannot detect a unidirectional link if it is connected to a UDLD-incapable port of another switch.
- When configuring the mode (normal or aggressive), make sure that the same mode is configured on both sides of the link.

Enabling UDLD Globally

Beginning in privileged EXEC mode, follow these steps to enable UDLD in the aggressive or normal mode and to set the configurable message timer on all fiber-optic ports on the switch:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**udld** {**aggressive** \| **enable** \| **message time** *message-timer-interval*}	Specify the UDLD mode of operation: • **aggressive**—Enables UDLD in aggressive mode on all fiber-optic ports. • **enable**—Enables UDLD in normal mode on all fiber-optic ports on the switch. UDLD is disabled by default. An individual interface configuration overrides the setting of the **udld enable** global configuration command. For more information about aggressive and normal modes, see the "Modes of Operation" section on page 23-1. • **message time** *message-timer-interval*—Configures the period of time between UDLD probe messages on ports that are in the advertisement phase and are detected to be bidirectional. The range is from 7 to 90 seconds. **Note** This command affects fiber-optic ports only. Use the **udld** interface configuration command to enable UDLD on other port types. For more information, see the "Enabling UDLD on an Interface" section on page 23-5.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show udld**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable UDLD globally, use the **no udld enable** global configuration command to disable normal mode UDLD on all fiber-optic ports. Use the **no udld aggressive** global configuration command to disable aggressive mode UDLD on all fiber-optic ports.

Enabling UDLD on an Interface

Beginning in privileged EXEC mode, follow these steps either to enable UDLD in the aggressive or normal mode or to disable UDLD on a port:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the port to be enabled for UDLD, and enter interface configuration mode.
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled, and NNIs are enabled.

	Command	Purpose
Step 4	**udld port** [**aggressive**]	UDLD is disabled by default. • **udld port**—Enables UDLD in normal mode on the specified port. • **udld port aggressive**—Enables UDLD in aggressive mode on the specified port. **Note** Use the **no udld port** interface configuration command to disable UDLD on a specified fiber-optic port. For more information about aggressive and normal modes, see the "Modes of Operation" section on page 23-1.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show udld** *interface-id*	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Resetting an Interface Disabled by UDLD

Beginning in privileged EXEC mode, follow these steps to reset all ports disabled by UDLD:

	Command	Purpose
Step 1	**udld reset**	Reset all ports disabled by UDLD.
Step 2	**show udld**	Verify your entries.

You can also bring up the port by using these commands:

- The **shutdown** interface configuration command followed by the **no shutdown** interface configuration command restarts the disabled port.
- The **no udld** {**aggressive** | **enable**} global configuration command followed by the **udld** {**aggressive** | **enable**} global configuration command re-enables the disabled ports.
- The **no udld port** interface configuration command followed by the **udld port** [**aggressive**] interface configuration command re-enables the disabled fiber-optic port.
- The **errdisable recovery cause udld** global configuration command enables the timer to automatically recover from the UDLD error-disabled state, and the **errdisable recovery interval** *interval* global configuration command specifies the time to recover from the UDLD error-disabled state.

Displaying UDLD Status

To display the UDLD status for the specified port or for all ports, use the **show udld** [*interface-id*] privileged EXEC command.

For detailed information about the fields in the command output, see the command reference for this release.



Configuring SPAN and RSPAN

This chapter describes how to configure Switched Port Analyzer (SPAN) and Remote SPAN (RSPAN) on the Cisco ME 3400 Ethernet Access switch.

Note For complete syntax and usage information for the commands used in this chapter, see the command reference for this release.

This chapter consists of these sections:

- Understanding SPAN and RSPAN, page 24-1
- Configuring SPAN and RSPAN, page 24-9
- Displaying SPAN and RSPAN Status, page 24-22

Understanding SPAN and RSPAN

You can analyze network traffic passing through ports or VLANs by using SPAN or RSPAN to send a copy of the traffic to another port on the switch or on another switch that has been connected to a network analyzer or other monitoring or security device. SPAN copies (or mirrors) traffic received or sent (or both) on source ports or source VLANs to a destination port for analysis. SPAN does not affect the switching of network traffic on the source ports or VLANs. You must dedicate the destination port for SPAN use. Except for traffic that is required for the SPAN or RSPAN session, destination ports do not receive or forward traffic.

Only traffic that enters or leaves source ports or traffic that enters or leaves source VLANs can be monitored by using SPAN; traffic routed to a source VLAN cannot be monitored. For example, if incoming traffic is being monitored, traffic that gets routed from another VLAN to the source VLAN cannot be monitored; however, traffic that is received on the source VLAN and routed to another VLAN can be monitored.

You can use the SPAN or RSPAN destination port to inject traffic from a network security device. For example, if you connect a Cisco Intrusion Detection System (IDS) sensor appliance to a destination port, the IDS device can send TCP reset packets to close down the TCP session of a suspected attacker.

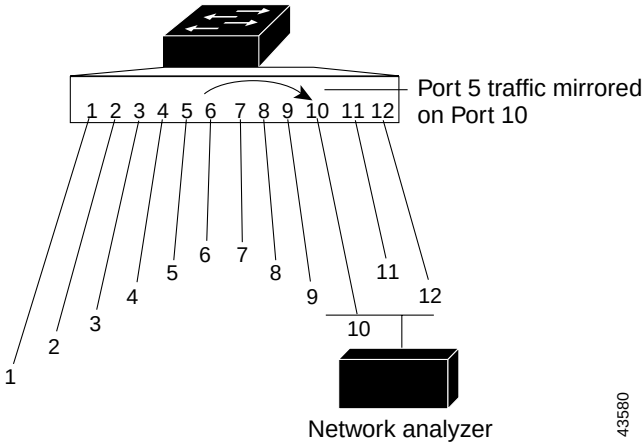
These sections contain this conceptual information:

- Local SPAN, page 24-2
- Remote SPAN, page 24-2
- SPAN and RSPAN Concepts and Terminology, page 24-3
- SPAN and RSPAN Interaction with Other Features, page 24-8

Local SPAN

Local SPAN supports a SPAN session entirely within one switch; all source ports or source VLANs and destination ports reside in the same switch. Local SPAN copies traffic from one or more source ports in any VLAN or from one or more VLANs to a destination port for analysis. For example, in Figure 24-1, all traffic on port 5 (the source port) is mirrored to port 10 (the destination port). A network analyzer on port 10 receives all network traffic from port 5 without being physically attached to port 5.

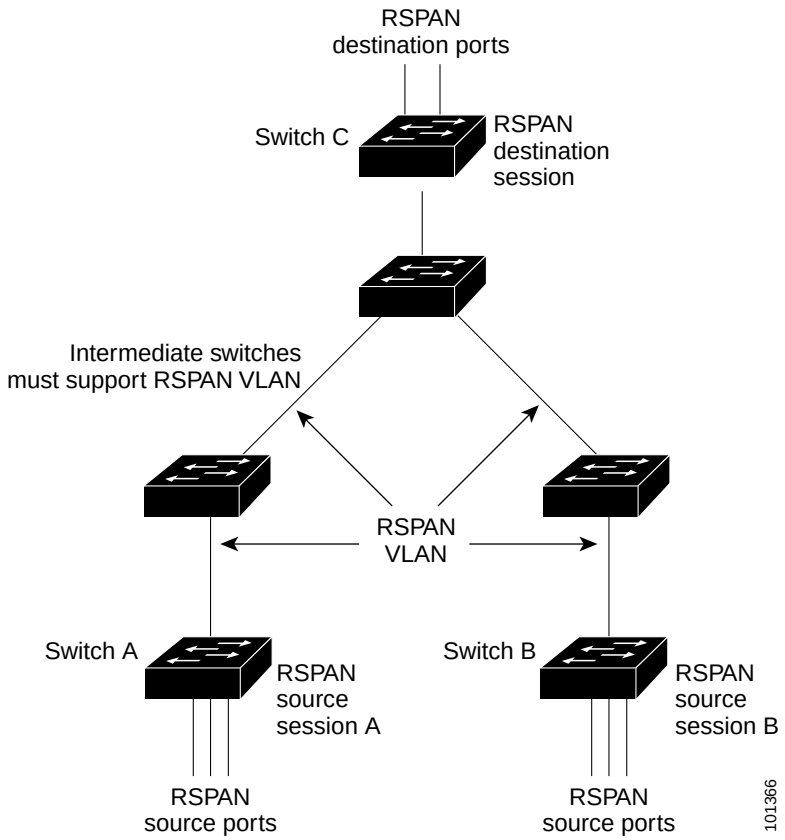
Figure 24-1 Example of Local SPAN Configuration on a Single Switch



Remote SPAN

RSPAN supports source ports, source VLANs, and destination ports on different switches, enabling remote monitoring of multiple switches across your network. Figure 24-2 shows source ports on Switch A and Switch B. The traffic for each RSPAN session is carried over a user-specified RSPAN VLAN that is dedicated for that RSPAN session in all participating switches. The RSPAN traffic from the source ports or VLANs is copied into the RSPAN VLAN and forwarded over trunk ports carrying the RSPAN VLAN to a destination session monitoring the RSPAN VLAN. Each RSPAN source switch must have either ports or VLANs as RSPAN sources. The destination is always a physical port, as shown on Switch C in the figure.

Figure 24-2 Example of RSPAN Configuration



SPAN and RSPAN Concepts and Terminology

This section describes concepts and terminology associated with SPAN and RSPAN configuration.

SPAN Sessions

SPAN sessions (local or remote) allow you to monitor traffic on one or more ports, or one or more VLANs, and send the monitored traffic to one or more destination ports.

A local SPAN session is an association of a destination port with source ports or source VLANs, all on a single network device. Local SPAN does not have separate source and destination sessions. Local SPAN sessions gather a set of ingress and egress packets specified by the user and form them into a stream of SPAN data, which is directed to the destination port.

RSPAN consists of at least one RSPAN source session, an RSPAN VLAN, and at least one RSPAN destination session. You separately configure RSPAN source sessions and RSPAN destination sessions on different network devices. To configure an RSPAN source session on a device, you associate a set of source ports or source VLANs with an RSPAN VLAN. The output of this session is the stream of SPAN packets that are sent to the RSPAN VLAN. To configure an RSPAN destination session on another device, you associate the destination port with the RSPAN VLAN. The destination session collects all RSPAN VLAN traffic and sends it out the RSPAN destination port.

An RSPAN source session is very similar to a local SPAN session, except for where the packet stream is directed. In an RSPAN source session, SPAN packets are relabeled with the RSPAN VLAN ID and directed over normal trunk ports to the destination switch.

An RSPAN destination session takes all packets received on the RSPAN VLAN, strips off the VLAN tagging, and presents them on the destination port. Its purpose is to present a copy of all RSPAN VLAN packets (except Layer 2 control packets) to the user for analysis.

There can be more than one source session and more than one destination session active in the same RSPAN VLAN. There can also be intermediate switches separating the RSPAN source and destination sessions. These switches need not be capable of running RSPAN, but they must respond to the requirements of the RSPAN VLAN (see the "RSPAN VLAN" section on page 24-8).

Traffic monitoring in a SPAN session has these restrictions:

- Sources can be ports or VLANs, but you cannot mix source ports and source VLANs in the same session.
- The switch supports up to two source sessions (local SPAN and RSPAN source sessions). You can run both a local SPAN and an RSPAN source session in the same switch. The switch supports a total of 66 source and RSPAN destination sessions.
- You can have multiple destination ports in a SPAN session, but no more than 64 destination ports.
- You can configure two separate SPAN or RSPAN source sessions with separate or overlapping sets of SPAN source ports and VLANs. When the metro IP access image in running on the switch, both switched and routed ports can be configured as SPAN sources and destinations.
- SPAN sessions do not interfere with the normal operation of the switch. However, an oversubscribed SPAN destination, for example, a 10-Mbps port monitoring a 100-Mbps port, can result in dropped or lost packets.
- When RSPAN is enabled, each packet being monitored is transmitted twice, once as normal traffic and once as a monitored packet. Therefore monitoring a large number of ports or VLANs could potentially generate large amounts of network traffic.
- You can configure SPAN sessions on disabled ports; however, a SPAN session does not become active unless you enable the destination port and at least one source port or VLAN for that session.
- The switch does not support a combination of local SPAN and RSPAN in a single session. That is, an RSPAN source session cannot have a local destination port, an RSPAN destination session cannot have a local source port, and an RSPAN destination session and an RSPAN source session that are using the same RSPAN VLAN cannot run on the same switch.

Monitored Traffic

SPAN sessions can monitor these traffic types:

- Receive (Rx) SPAN—The goal of receive (or ingress) SPAN is to monitor as much as possible all the packets received by the source interface or VLAN before any modification or processing is performed by the switch. A copy of each packet received by the source is sent to the destination port for that SPAN session.

 Packets that are modified because of routing or quality of service (QoS)—for example, modified Differentiated Services Code Point (DSCP)—are copied before modification.

 Features that can cause a packet to be dropped during receive processing have no effect on ingress SPAN; the destination port receives a copy of the packet even if the actual incoming packet is dropped. These features include IP standard and extended input access control lists (ACLs), ingress QoS policing, VLAN ACLs and egress QoS policing.

- Transmit (Tx) SPAN—The goal of transmit (or egress) SPAN is to monitor as much as possible all the packets sent by the source interface after all modification and processing is performed by the switch. A copy of each packet sent by the source is sent to the destination port for that SPAN session. The copy is provided after the packet is modified.

 Packets that are modified because of routing—for example, with modified time-to-live (TTL), MAC-address, or QoS values—are duplicated (with the modifications) at the destination port.

 Features that can cause a packet to be dropped during transmit processing also affect the duplicated copy for SPAN. These features include IP standard and extended output ACLs and egress QoS policing.

- Both—In a SPAN session, you can also monitor a port or VLAN for both received and sent packets. This is the default.

The default configuration for local SPAN session ports is to send all packets untagged. SPAN also does not normally monitor bridge protocol data unit (BPDU) packets and Layer 2 protocols, such as Cisco Discovery Protocol (CDP), Spanning Tree Protocol (STP), and Port Aggregation Protocol (PAgP). However, when you enter the **encapsulation replicate** keywords when configuring a destination port, these changes occur:

- Packets are sent on the destination port with the same encapsulation—untagged or IEEE 802.1Q—that they had on the source port.
- Packets of all types, including BPDU and Layer 2 protocol packets, are monitored.

Therefore, a local SPAN session with encapsulation replicate enabled can have a mixture of untagged and IEEE 802.1Q tagged packets appear on the destination port.

Switch congestion can cause packets to be dropped at ingress source ports, egress source ports, or SPAN destination ports. In general, these characteristics are independent of one another. For example:

- A packet might be forwarded normally but dropped from monitoring due to an oversubscribed SPAN destination port.
- An ingress packet might be dropped from normal forwarding, but still appear on the SPAN destination port.
- An egress packet dropped because of switch congestion is also dropped from egress SPAN.

In some SPAN configurations, multiple copies of the same source packet are sent to the SPAN destination port. For example, a bidirectional (both Rx and Tx) SPAN session is configured for the Rx monitor on port A and Tx monitor on port B. If a packet enters the switch through port A and is switched to port B, both incoming and outgoing packets are sent to the destination port. Both packets are the same (unless a Layer-3 rewrite occurs, in which case the packets are different because of the packet modification).

Source Ports

A source port (also called a *monitored port*) is a switched or routed port that you monitor for network traffic analysis. In a local SPAN session or RSPAN source session, you can monitor source ports or VLANs for traffic in one or both directions. The switch supports any number of source ports (up to the maximum number of available ports on the switch) and any number of source VLANs (up to the maximum number of VLANs supported). However, the switch supports a maximum of two sessions (local or RSPAN) with source ports or VLANs, and you cannot mix ports and VLANs in a single session.

A source port has these characteristics:

- It can be monitored in multiple SPAN sessions.
- Each source port can be configured with a direction (ingress, egress, or both) to monitor.

- It can be any port type—for example, EtherChannel, Fast Ethernet, Gigabit Ethernet, user network interface (UNI), network node interface (NNI), and so forth.
- For EtherChannel sources, you can monitor traffic for the entire EtherChannel or individually on a physical port as it participates in the port channel.
- It can be a routed port, an access port, or a trunk port.
- It cannot be a destination port.
- Source ports can be in the same or different VLANs.
- You can monitor multiple source ports in a single session.

Source VLANs

VLAN-based SPAN (VSPAN) is the monitoring of the network traffic in one or more VLANs. The SPAN or RSPAN source interface in VSPAN is a VLAN ID, and traffic is monitored on all the ports for that VLAN.

VSPAN has these characteristics:

- All active ports in the source VLAN are included as source ports and can be monitored in either or both directions.
- On a given port, only traffic on the monitored VLAN is sent to the destination port.
- If a destination port belongs to a source VLAN, it is excluded from the source list and is not monitored.
- If ports are added to or removed from the source VLANs, the traffic on the source VLAN received by those ports is added to or removed from the sources being monitored.
- You cannot use filter VLANs in the same session with VLAN sources.
- You can monitor only Ethernet VLANs.

VLAN Filtering

When you monitor a trunk port as a source port, by default, all VLANs active on the trunk are monitored. You can limit SPAN traffic monitoring on trunk source ports to specific VLANs by using VLAN filtering.

- VLAN filtering applies only to trunk ports.
- VLAN filtering applies only to port-based sessions and is not allowed in sessions with VLAN sources.
- When a VLAN filter list is specified, only those VLANs in the list are monitored on trunk ports.
- SPAN traffic coming from other port types is not affected by VLAN filtering; that is, all VLANs are allowed on other ports.
- VLAN filtering affects only traffic forwarded to the destination SPAN port and does not affect the switching of normal traffic.

Destination Port

Each local SPAN session or RSPAN destination session must have a destination port (also called a *monitoring port*) that receives a copy of traffic from the source ports or VLANs and sends the SPAN packets to the user, usually a network analyzer.

A destination port has these characteristics:

- For a local SPAN session, the destination port must reside on the same switch as the source port. For an RSPAN session, it is located on the switch containing the RSPAN destination session. There is no destination port on a switch running only an RSPAN source session.
- When a port is configured as a SPAN destination port, the configuration overwrites the original port configuration. When the SPAN destination configuration is removed, the port reverts to its previous configuration. If a configuration change is made to the port while it is acting as a SPAN destination port, the change does not take effect until the SPAN destination configuration had been removed.
- If the port was in an EtherChannel group, it is removed from the group while it is a destination port. If the switch is running the metro IP access image and the port was a routed port, it is no longer a routed port.
- It can be any Ethernet physical port.
- It cannot be a secure port.
- It cannot be a source port.
- It cannot be an EtherChannel group or a VLAN.
- It can participate in only one SPAN session at a time (a destination port in one SPAN session cannot be a destination port for a second SPAN session).
- When it is active, incoming traffic is disabled. The port does not send any traffic except that required for the SPAN session. Incoming traffic is never learned or forwarded on a destination port.
- If incoming traffic forwarding is enabled for a network security device, the destination port forwards traffic at Layer 2.
- It does not participate in any of the Layer 2 protocols (STP, VTP, CDP, DTP, PagP).
- A destination port that belongs to a source VLAN of any SPAN session is excluded from the source list and is not monitored.
- The maximum number of destination ports in a switch is 64.

Local SPAN and RSPAN destination ports behave differently regarding VLAN tagging and encapsulation:

- For local SPAN, if the **encapsulation replicate** keywords are specified for the destination port, these packets appear with the original encapsulation (untagged or IEEE 802.1Q). If **encapsulation dot1q** is specified, packets appear with the IEEE 802.1Q encapsulation. If these keywords are not specified, packets appear in the untagged format. Therefore, the output of a local SPAN session with **encapsulation replicate** enabled can contain a mixture of untagged and IEEE 802.1Q tagged packets.
- For RSPAN, the original VLAN ID is lost because it is overwritten by the RSPAN VLAN identification. Therefore, all packets appear on the destination port as untagged.

RSPAN VLAN

The RSPAN VLAN carries SPAN traffic between RSPAN source and destination sessions. It has these special characteristics:

- All traffic in the RSPAN VLAN is always flooded.
- No MAC address learning occurs on the RSPAN VLAN.
- RSPAN VLAN traffic only flows on trunk ports.
- RSPAN VLANs must be configured in VLAN configuration mode by using the **remote-span** VLAN configuration mode command.
 - To change a VLAN from a UNI isolated VLAN (the default) to an RSPAN VLAN, enter the **rspan-vlan** VLAN configuration command.
 - To change a UNI community VLAN to an RSPAN VLAN, you must first remove the community VLAN type by entering the **no uni-vlan** VLAN configuration command.
- STP can run on RSPAN VLAN trunks but not on SPAN destination ports.

Note Only NNIs support STP; UNIs do not support STP.

- An RSPAN VLAN cannot be a private-VLAN primary or secondary VLAN.

It is normal to have multiple RSPAN VLANs in a network at the same time with each RSPAN VLAN defining a network-wide RSPAN session. That is, multiple RSPAN source sessions anywhere in the network can contribute packets to the RSPAN session. It is also possible to have multiple RSPAN destination sessions throughout the network, monitoring the same RSPAN VLAN and presenting traffic to the user. The RSPAN VLAN ID separates the sessions.

SPAN and RSPAN Interaction with Other Features

SPAN interacts with these features:

- Routing—For switches that are running the metro IP access image, SPAN does not monitor routed traffic. RSPAN only monitors traffic that enters or exits the switch, not traffic that is routed between VLANs. For example, if a VLAN is being receive-monitored and the switch routes traffic from another VLAN to the monitored VLAN, that traffic is not monitored and is not received on the SPAN destination port.
- STP—A destination port does not participate in STP while its SPAN or RSPAN session is active. The destination port can participate in STP after the SPAN or RSPAN session is disabled. On a source port, SPAN does not affect the STP status. STP can be active on trunk ports carrying an RSPAN VLAN. However, only NNIs support STP; UNIs do not participate in STP.
- CDP—A SPAN destination port does not participate in CDP while the SPAN session is active. After the SPAN session is disabled, the port again participates in CDP. However, only NNIs support CDP; UNIs do not participate in CDP.
- VLAN and trunking—You can modify VLAN membership or trunk settings for source or destination ports at any time. However, changes in VLAN membership or trunk settings for a destination port do not take effect until you remove the SPAN destination configuration. Changes in VLAN membership or trunk settings for a source port immediately take effect, and the respective SPAN sessions automatically adjust accordingly.

- EtherChannel—You can configure an EtherChannel group as a source port but not as a SPAN destination port. When a group is configured as a SPAN source, the entire group is monitored.

 If a physical port is added to a monitored EtherChannel group, the new port is added to the SPAN source port list. If a port is removed from a monitored EtherChannel group, it is automatically removed from the source port list.

 A physical port that belongs to an EtherChannel group can be configured as a SPAN source port and still be a part of the EtherChannel. In this case, data from the physical port is monitored as it participates in the EtherChannel. However, if a physical port that belongs to an EtherChannel group is configured as a SPAN destination, it is removed from the group. After the port is removed from the SPAN session, it rejoins the EtherChannel group. Ports removed from an EtherChannel group remain members of the group, but they are in the *inactive* or *suspended* state.

 If a physical port that belongs to an EtherChannel group is a destination port and the EtherChannel group is a source, the port is removed from the EtherChannel group and from the list of monitored ports.
- Multicast traffic can be monitored. For sending and receiving port monitoring, only a single unedited packet is sent to the SPAN destination port. It does not reflect the number of times that the multicast packet is sent.
- A private-VLAN port cannot be a SPAN destination port.
- A secure port cannot be a SPAN destination port.

 For SPAN sessions, do not enable port security on ports with monitored egress when ingress forwarding is enabled on the destination port. For RSPAN source sessions, do not enable port security on any ports with monitored egress.
- An IEEE 802.1x port can be a SPAN source port. You can enable IEEE 802.1x on a port that is a SPAN destination port; however, IEEE 802.1x is disabled until the port is removed as a SPAN destination.

 For SPAN sessions, do not enable IEEE 802.1x on ports with monitored sending when receive forwarding is enabled on the destination port. For RSPAN source sessions, do not enable IEEE 802.1x on any ports that are sending monitored.

Configuring SPAN and RSPAN

These sections contain this configuration information:

- Default SPAN and RSPAN Configuration, page 24-10
- Configuring Local SPAN, page 24-10
- Configuring RSPAN, page 24-16

Default SPAN and RSPAN Configuration

Table 24-1 shows the default SPAN and RSPAN configuration.

Table 24-1 Default SPAN and RSPAN Configuration

Feature	Default Setting
SPAN state (SPAN and RSPAN)	Disabled.
Source port traffic to monitor	Both received and sent traffic (**both**).
Encapsulation type (destination port)	Native form (untagged packets).
Ingress forwarding (destination port)	Disabled
VLAN filtering	On a trunk interface used as a source port, all VLANs are monitored.
RSPAN VLANs	None configured. Default VLAN type is UNI isolated.

Configuring Local SPAN

These sections contain this configuration information:

- SPAN Configuration Guidelines, page 24-10
- Creating a Local SPAN Session, page 24-11
- Creating a Local SPAN Session and Configuring Ingress Traffic, page 24-13
- Specifying VLANs to Filter, page 24-15

SPAN Configuration Guidelines

Follow these guidelines when configuring SPAN:

- You can configure a total of two local SPAN sessions or RSPAN source sessions on each switch. You can have a total of 66 SPAN sessions (local, RSPAN source, and RSPAN destination) on a switch.
- For SPAN sources, you can monitor traffic for a single port or VLAN or a series or range of ports or VLANs for each session. You cannot mix source ports and source VLANs within a single SPAN session.
- The destination port cannot be a source port; a source port cannot be a destination port.
- You cannot have two SPAN sessions using the same destination port.
- When you configure a switch port as a SPAN destination port, it is no longer a normal switch port; only monitored traffic passes through the SPAN destination port.
- Entering SPAN configuration commands does not remove previously configured SPAN parameters. You must enter the **no monitor session** {*session_number* | **all** | **local** | **remote**} global configuration command to delete configured SPAN parameters.
- For local SPAN, outgoing packets through the SPAN destination port carry the original encapsulation headers—untagged or IEEE 802.1Q—if the **encapsulation replicate** or **encapsulation dot1q** keywords are specified. If the keywords are not specified, the packets are sent in native form. For RSPAN destination ports, outgoing packets are not tagged.
- You can configure a disabled port to be a source or destination port, but the SPAN function does not start until the destination port and at least one source port or source VLAN are enabled.

- You can limit SPAN traffic to specific VLANs by using the **filter vlan** keyword. If a trunk port is being monitored, only traffic on the VLANs specified with this keyword is monitored. By default, all VLANs are monitored on a trunk port.
- You cannot mix source VLANs and filter VLANs within a single SPAN session.

Creating a Local SPAN Session

Beginning in privileged EXEC mode, follow these steps to create a SPAN session and specify the source (monitored) ports or VLANs and the destination (monitoring) ports:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**no monitor session** {*session_number* \| **all** \| **local** \| **remote**}	Remove any existing SPAN configuration for the session. For *session_number*, the range is 1 to 66. Specify **all** to remove all SPAN sessions, **local** to remove all local sessions, or **remote** to remove all remote SPAN sessions.
Step 3	**monitor session** *session_number* **source** {**interface** *interface-id* \| **vlan** *vlan-id*} [**,** \| **-**] [**both** \| **rx** \| **tx**]	Specify the SPAN session and the source port (monitored port). For *session_number*, the range is 1 to 66. For *interface-id*, specify the source port or source VLAN to monitor. • For source *interface-id*, specify the source port to monitor. Valid interfaces include physical interfaces and port-channel logical interfaces (**port-channel** *port-channel-number*). Valid port-channel numbers are 1 to 48. • For *vlan-id*, specify the source VLAN to monitor. The range is 1 to 4094 (excluding the RSPAN VLAN). **Note** A single session can include multiple sources (ports or VLANs), defined in a series of commands, but you cannot combine source ports and source VLANs in one session. (Optional) [**,** \| **-**] Specify a series or range of interfaces. Enter a space before and after the comma; enter a space before and after the hyphen. (Optional) Specify the direction of traffic to monitor. If you do not specify a traffic direction, the SPAN monitors both sent and received traffic. • **both**—Monitor both received and sent traffic. This is the default. • **rx**—Monitor received traffic. • **tx**—Monitor sent traffic. **Note** You can use the **monitor session** *session_number* **source** command multiple times to configure multiple source ports.

	Command	Purpose
Step 4	**monitor session** *session_number* **destination** {**interface** *interface-id* [, \| -] [**encapsulation**{**dot1q** \| **replicate**}]	Specify the SPAN session and the destination port (monitoring port). For *session_number*, specify the session number entered in Step 3. **Note** For local SPAN, you must use the same session number for the source and destination interfaces. For *interface-id*, specify the destination port. The destination interface must be a physical port; it cannot be an EtherChannel, and it cannot be a VLAN. (Optional) [**,** \| -] Specify a series or range of interfaces. Enter a space before and after the comma; enter a space before and after the hyphen. (Optional) Enter **encapsulation dot1q** for IEEE 802.1Q encapsulation or **encapsulation replicate** to specify that the destination interface replicates the source interface encapsulation method. If not selected, the default is to send packets in native form (untagged). **Note** You can use **monitor session** *session_number* **destination** command multiple times to configure multiple destination ports.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show monitor** [**session** *session_number*] **show running-config**	Verify the configuration.
Step 7	**copy running-config startup-config**	(Optional) Save the configuration in the configuration file.

To delete a SPAN session, use the **no monitor session** *session_number* global configuration command. To remove a source or destination port or VLAN from the SPAN session, use the **no monitor session** *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*} global configuration command or the **no monitor session** *session_number* **destination interface** *interface-id* global configuration command. For destination interfaces, the **encapsulation replicate** keywords are ignored with the **no** form of the command.

This example shows how to set up SPAN session 1 for monitoring source port traffic to a destination port. First, any existing SPAN configuration for session 1 is deleted, and then bidirectional traffic is mirrored from source Gigabit Ethernet port 1 to destination Gigabit Ethernet port 2, retaining the encapsulation method.

```
Switch(config)# no monitor session 1
Switch(config)# monitor session 1 source interface gigabitethernet0/1
Switch(config)# monitor session 1 destination interface gigabitethernet0/2
encapsulation replicate
Switch(config)# end
```

This example shows how to remove port 1 as a SPAN source for SPAN session 1:

```
Switch(config)# no monitor session 1 source interface gigabitethernet0/1
Switch(config)# end
```

This example shows how to disable received traffic monitoring on port 1, which was configured for bidirectional monitoring:

```
Switch(config)# no monitor session 1 source interface gigabitethernet0/1 rx
```

The monitoring of traffic received on port 1 is disabled, but traffic sent from this port continues to be monitored.

This example shows how to remove any existing configuration on SPAN session 2, configure SPAN session 2 to monitor received traffic on all ports belonging to VLANs 1 through 3, and send it to destination Gigabit Ethernet port 2. The configuration is then modified to also monitor all traffic on all ports belonging to VLAN 10.

```
Switch(config)# no monitor session 2
Switch(config)# monitor session 2 source vlan 1 - 3 rx
Switch(config)# monitor session 2 destination interface gigabitethernet0/2
Switch(config)# monitor session 2 source vlan 10
Switch(config)# end
```

Creating a Local SPAN Session and Configuring Ingress Traffic

Beginning in privileged EXEC mode, follow these steps to create a SPAN session, to specify the source ports or VLANs and the destination ports, and to enable ingress traffic on the destination port for a network security device (such as a Cisco IDS Sensor Appliance).

Note See the "Creating a Local SPAN Session" section on page 24-11 for details about the keywords not related to ingress traffic.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**no monitor session** {*session_number* \| **all** \| **local** \| **remote**}	Remove any existing SPAN configuration for the session.
Step 3	**monitor session** *session_number* **source** {**interface** *interface-id* \| **vlan** *vlan-id*} [**,** \| **-**] [**both** \| **rx** \| **tx**]	Specify the SPAN session and the source port (monitored port).

	Command	Purpose
Step 4	**monitor session** *session_number* **destination** {**interface** *interface-id* [, \| -] [**encapsulation** {**dot1q \| replicate}**] [**ingress** {[**dot1q \| untagged] vlan** *vlan-id*}]	Specify the SPAN session, the destination port, the packet encapsulation, and the ingress VLAN and encapsulation. For *session_number*, specify the session number entered in Step 3. For *interface-id*, specify the destination port. The destination interface must be a physical port; it cannot be an EtherChannel, and it cannot be a VLAN. (Optional) [**,** \| -] Specify a series or range of interfaces. Enter a space before and after the comma or hyphen. (Optional) Enter **encapsulation dot1q** for IEEE 802.1Q encapsulation or **encapsulation replicate** to specify that the destination interface replicates the source interface encapsulation method. If not selected, the default is to send packets in native form (untagged). Enter **ingress** with keywords to enable ingress traffic forwarding on the destination port and specify the encapsulation type: • **dot1q**—Forward incoming packets with IEEE 802.1Q encapsulation with the specified VLAN as the default VLAN. • **untagged**—Forward incoming packets with untagged encapsulation type with the specified VLAN as the default VLAN. • **vlan** *vlan-id*—The default VLAN. If neither **dot1q** or **untagged** is specified, the default is to forward packets untagged.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show monitor** [**session** *session_number*] **show running-config**	Verify the configuration.
Step 7	**copy running-config startup-config**	(Optional) Save the configuration in the configuration file.

To delete a SPAN session, use the **no monitor session** *session_number* global configuration command. To remove a source or destination port or VLAN from the SPAN session, use the **no monitor session** *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*} global configuration command or the **no monitor session** *session_number* **destination interface** *interface-id* global configuration command. For destination interfaces, the encapsulation and ingress options are ignored with the **no** form of the command.

This example shows how to remove any existing configuration on SPAN session 2, configure SPAN session 2 to monitor sent traffic on Gigabit Ethernet source port 1, and send it to destination Gigabit Ethernet port 2 with the same egress encapsulation type as the source port, and to enable incoming forwarding with IEEE 802.1Q encapsulation and VLAN 6 as the default ingress VLAN.

```
Switch(config)# no monitor session 2
Switch(config)# monitor session 2 source gigabitethernet0/1 rx
Switch(config)# monitor session 2 destination interface gigabitethernet0/2 encapsulation
replicate ingress dot1q vlan 6
Switch(config)# end
```

Specifying VLANs to Filter

Beginning in privileged EXEC mode, follow these steps to limit SPAN source traffic to specific VLANs:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**no monitor session** {*session_number* \| **all** \| **local** \| **remote**}	Remove any existing SPAN configuration for the session. For *session_number*, the range is 1 to 66. Specify **all** to remove all SPAN sessions, **local** to remove all local sessions, or **remote** to remove all remote SPAN sessions.
Step 3	**monitor session** *session_number* **source interface** *interface-id*	Specify the characteristics of the source port (monitored port) and SPAN session. For *session_number*, the range is 1 to 66. For *interface-id*, specify the source port to monitor. The interface specified must already be configured as a trunk port.
Step 4	**monitor session** *session_number* **filter vlan** *vlan-id* [**,** \| -]	Limit the SPAN source traffic to specific VLANs. For *session_number*, enter the session number specified in Step 3. For *vlan-id*, the range is 1 to 4094. (Optional) Use a comma (**,**) to specify a series of VLANs, or use a hyphen (**-**) to specify a range of VLANs. Enter a space before and after the comma; enter a space before and after the hyphen.
Step 5	**monitor session** *session_number* **destination** {**interface** *interface-id* [, \| -] [**encapsulation** {**dot1q** \| **replicate}**]}	Specify the SPAN session and the destination port (monitoring port). For *session_number*, specify the session number entered in Step 3. For *interface-id*, specify the destination port. The destination interface must be a physical port; it cannot be an EtherChannel, and it cannot be a VLAN. (Optional) [**,** \| -] Specify a series or range of interfaces. Enter a space before and after the comma; enter a space before and after the hyphen. (Optional) Enter **encapsulation dot1q** or **encapsulation replicate** to specify IEEE 802.1Q encapsulation or that the destination interface replicates the source interface encapsulation method. If not selected, the default is to send packets in native form (untagged).
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show monitor** [**session** *session_number*] **show running-config**	Verify the configuration.
Step 8	**copy running-config startup-config**	(Optional) Save the configuration in the configuration file.

To monitor all VLANs on the trunk port, use the **no monitor session** *session_number* **filter** global configuration command.

This example shows how to remove any existing configuration on SPAN session 2, configure SPAN session 2 to monitor traffic received on Gigabit Ethernet trunk port 2, and send traffic for only VLANs 1 through 5 and VLAN 9 to destination Gigabit Ethernet port 1.

```
Switch(config)# no monitor session 2
Switch(config)# monitor session 2 source interface gigabitethernet0/2 rx
Switch(config)# monitor session 2 filter vlan 1 - 5 , 9
Switch(config)# monitor session 2 destination interface gigabitethernet0/1
Switch(config)# end
```

Configuring RSPAN

These sections contain this configuration information:

- RSPAN Configuration Guidelines, page 24-16
- Configuring a VLAN as an RSPAN VLAN, page 24-17
- Creating an RSPAN Source Session, page 24-17
- Creating an RSPAN Destination Session, page 24-19
- Creating an RSPAN Destination Session and Configuring Ingress Traffic, page 24-20
- Specifying VLANs to Filter, page 24-21

RSPAN Configuration Guidelines

Follow these guidelines when configuring RSPAN:

- All the items in the "SPAN Configuration Guidelines" section on page 24-10 apply to RSPAN.
- As RSPAN VLANs have special properties, you should reserve a few VLANs across your network for use as RSPAN VLANs; do not assign access ports to these VLANs.
- You can apply an output ACL to RSPAN traffic to selectively filter or monitor specific packets. Specify these ACLs on the RSPAN VLAN in the RSPAN source switches.
- For RSPAN configuration, you can distribute the source ports and the destination ports across multiple switches in your network.
- RSPAN does not support BPDU packet monitoring or other Layer 2 switch protocols.
- The RSPAN VLAN is configured only on trunk ports and not on access ports. To avoid unwanted traffic in RSPAN VLANs, make sure that the VLAN remote-span feature is supported in all the participating switches.
- Access ports on the RSPAN VLAN are put in the inactive state.
- RSPAN VLANs are included as sources for port-based RSPAN sessions when source trunk ports have active RSPAN VLANs. RSPAN VLANs can also be sources in SPAN sessions. However, since the switch does not monitor spanned traffic, it does not support egress spanning of packets on any RSPAN VLAN identified as the destination of an RSPAN source session on the switch.
- You can configure any VLAN as an RSPAN VLAN as long as these conditions are met:
 - The same RSPAN VLAN is used for an RSPAN session in all the switches.
 - All participating switches support RSPAN.
 - MAC address learning is not disabled on the RSPAN VLAN.
- We recommend that you configure an RSPAN VLAN before you configure an RSPAN source or a destination session.

Configuring a VLAN as an RSPAN VLAN

Create a new VLAN to be the RSPAN VLAN for the RSPAN session. You must create the RSPAN VLAN in all switches that will participate in RSPAN. You must configure RSPAN VLAN on source and destination switches and any intermediate switches.

To get an efficient flow of RSPAN traffic, manually delete the RSPAN VLAN from all trunks that do not need to carry the RSPAN traffic.

Beginning in privileged EXEC mode, follow these steps to create an RSPAN VLAN:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**vlan** *vlan-id*	Enter a VLAN ID to create a VLAN, or enter the VLAN ID of an existing VLAN, and enter VLAN configuration mode. The range is 2 to 1001 and 1006 to 4094. **Note** The RSPAN VLAN cannot be VLAN 1 (the default VLAN) or VLAN IDs 1002 through 1005 (reserved for Token Ring and FDDI VLANs). If you enter the VLAN ID of a UNI community VLAN, you must remove the community VLAN type by entering the **no uni-vlan** VLAN configuration command.
Step 3	**remote-span**	Configure the VLAN as an RSPAN VLAN.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**copy running-config startup-config**	(Optional) Save the configuration in the configuration file.

To remove the remote SPAN characteristic from a VLAN and convert it back to a UNI isolated VLAN, use the **no remote-span** VLAN configuration command.

This example shows how to create RSPAN VLAN 901.

```
Switch(config)# vlan 901
Switch(config-vlan)# remote span
Switch(config-vlan)# end
```

Creating an RSPAN Source Session

Beginning in privileged EXEC mode, follow these steps to start an RSPAN source session and to specify the monitored source and the destination RSPAN VLAN:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**no monitor session** {*session_number* \| **all** \| **local** \| **remote**}	Remove any existing RSPAN configuration for the session. For *session_number*, the range is 1 to 66. Specify **all** to remove all RSPAN sessions, **local** to remove all local sessions, or **remote** to remove all remote SPAN sessions.

	Command	Purpose
Step 3	**monitor session** *session_number* **source** {**interface** *interface-id* \| **vlan** *vlan-id*} [**,** \| **-**] [**both** \| **rx** \| **tx**]	Specify the RSPAN session and the source port (monitored port). For *session_number*, the range is 1 to 66. Enter a source port or source VLAN for the RSPAN session: • For *interface-id*, specify the source port to monitor. Valid interfaces include physical interfaces and port-channel logical interfaces (**port-channel** *port-channel-number*). Valid port-channel numbers are 1 to 48. • For *vlan-id*, specify the source VLAN to monitor. The range is 1 to 4094 (excluding the RSPAN VLAN). **Note** A single session can include multiple sources (ports or VLANs), defined in a series of commands, but you cannot combine source ports and source VLANs in one session. (Optional) [**,** \| **-**] Specify a series or range of interfaces. Enter a space before and after the comma; enter a space before and after the hyphen. (Optional) Specify the direction of traffic to monitor. If you do not specify a traffic direction, the source interface sends both sent and received traffic. • **both**—Monitor both received and sent traffic. • **rx**—Monitor received traffic. • **tx**—Monitor sent traffic.
Step 4	**monitor session** *session_number* **destination remote vlan** *vlan-id*	Specify the RSPAN session and the destination RSPAN VLAN. For *session_number*, enter the number defined in Step 3. For *vlan-id*, specify the source RSPAN VLAN to monitor.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show monitor** [**session** *session_number*] **show running-config**	Verify the configuration.
Step 7	**copy running-config startup-config**	(Optional) Save the configuration in the configuration file.

To delete a SPAN session, use the **no monitor session** *session_number* global configuration command.

To remove a source port or VLAN from the SPAN session, use the **no monitor session** *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*} global configuration command. To remove the RSPAN VLAN from the session, use the **no monitor session** *session_number* **destination remote vlan** *vlan-id*.

This example shows how to remove any existing RSPAN configuration for session 1, configure RSPAN session 1 to monitor multiple source interfaces, and configure the destination as RSPAN VLAN 901.

```
Switch(config)# no monitor session 1
Switch(config)# monitor session 1 source interface gigabitethernet0/1 tx
Switch(config)# monitor session 1 source interface gigabitethernet0/2 rx
Switch(config)# monitor session 1 source interface port-channel 12
Switch(config)# monitor session 1 destination remote vlan 901
Switch(config)# end
```

Creating an RSPAN Destination Session

You configure the RSPAN destination session on a different switch; that is, not the switch on which the source session was configured.

Beginning in privileged EXEC mode, follow these steps to define the RSPAN VLAN on that switch, to create an RSPAN destination session, and to specify the source RSPAN VLAN and the destination port:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**vlan** *vlan-id*	Enter the VLAN ID of the RSPAN VLAN created from the source switch, and enter VLAN configuration mode. **Note** If the VLAN is configured as a UNI community VLAN, you must remove the community VLAN type by entering the **no uni-vlan** VLAN configuration command.
Step 3	**remote-span**	Identify the VLAN as the RSPAN VLAN.
Step 4	**exit**	Return to global configuration mode.
Step 5	**no monitor session** {*session_number* \| **all** \| **local** \| **remote**}	Remove any existing RSPAN configuration for the session. For *session_number*, the range is 1 to 66. Specify **all** to remove all RSPAN sessions, **local** to remove all local sessions, or **remote** to remove all remote SPAN sessions.
Step 6	**monitor session** *session_number* **source remote vlan** *vlan-id*	Specify the RSPAN session and the source RSPAN VLAN. For *session_number*, the range is 1 to 66. For *vlan-id*, specify the source RSPAN VLAN to monitor.
Step 7	**monitor session** *session_number* **destination interface** *interface-id*	Specify the RSPAN session and the destination interface. For *session_number*, enter the number defined in Step 6. **Note** In an RSPAN destination session, you must use the same session number for the source RSPAN VLAN and the destination port. For *interface-id*, specify the destination interface. The destination interface must be a physical interface. **Note** Though visible in the command-line help string, **encapsulation replicate** is not supported for RSPAN. The original VLAN ID is overwritten by the RSPAN VLAN ID, and all packets appear on the destination port as untagged.
Step 8	**end**	Return to privileged EXEC mode.
Step 9	**show monitor** [**session** *session_number*] **show running-config**	Verify the configuration.
Step 10	**copy running-config startup-config**	(Optional) Save the configuration in the configuration file.

To delete a SPAN session, use the **no monitor session** *session_number* global configuration command. To remove a destination port from the SPAN session, use the **no monitor session** *session_number* **destination interface** *interface-id* global configuration command. To remove the RSPAN VLAN from the session, use the **no monitor session** *session_number* **source remote vlan** *vlan-id*.

This example shows how to configure VLAN 901 as the source remote VLAN and port 1 as the destination interface:

```
Switch(config)# monitor session 1 source remote vlan 901
Switch(config)# monitor session 1 destination interface gigabitethernet0/1
Switch(config)# end
```

Creating an RSPAN Destination Session and Configuring Ingress Traffic

Beginning in privileged EXEC mode, follow these steps to create an RSPAN destination session, to specify the source RSPAN VLAN and the destination port, and to enable ingress traffic on the destination port for a network security device (such as a Cisco IDS Sensor Appliance).

Note For details about the keywords not related to ingress traffic, see the "Creating an RSPAN Destination Session" section on page 24-19. This procedure assumes that the RSPAN VLAN has been configured.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**no monitor session** {*session_number* \| **all** \| **local** \| **remote**}	Remove any existing SPAN configuration for the session.
Step 3	**monitor session** *session_number* **source remote vlan** *vlan-id*	Specify the RSPAN session and the source RSPAN VLAN. For *session_number*, the range is 1 to 66. For *vlan-id*, specify the source RSPAN VLAN to monitor.
Step 4	**monitor session** *session_number* **destination** {**interface** *interface-id* [, \| -] [**ingress** {**dot1q vlan** *vlan-id* \| **untagged vlan** *vlan-id* \| **vlan** *vlan-id*}]}	Specify the SPAN session, the destination port, the packet encapsulation, and the ingress VLAN and encapsulation. For *session_number*, enter the number defined in Step 4. **Note** In an RSPAN destination session, you must use the same session number for the source RSPAN VLAN and the destination port. For *interface-id*, specify the destination interface. The destination interface must be a physical interface. **Note** Though visible in the command-line help string, **encapsulation replicate** is not supported for RSPAN. The original VLAN ID is overwritten by the RSPAN VLAN ID, and all packets appear on the destination port as untagged. (Optional) [**,** \| **-**] Specify a series or range of interfaces. Enter a space before and after the comma; enter a space before and after the hyphen. Enter **ingress** with additional keywords to enable ingress traffic forwarding on the destination port and to specify the encapsulation type: • **dot1q vlan** *vlan-id*—Forward incoming packets with IEEE 802.1Q encapsulation with the specified VLAN as the default VLAN. • **untagged vlan** *vlan-id* or **vlan** *vlan-id*—Forward incoming packets with untagged encapsulation type with the specified VLAN as the default VLAN.

	Command	Purpose
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show monitor** [**session** *session_number*] **show running-config**	Verify the configuration.
Step 7	**copy running-config startup-config**	(Optional) Save the configuration in the configuration file.

To delete an RSPAN session, use the **no monitor session** *session_number* global configuration command. To remove a destination port from the RSPAN session, use the **no monitor session** *session_number* **destination interface** *interface-id* global configuration command. The ingress options are ignored with the **no** form of the command.

This example shows how to configure VLAN 901 as the source remote VLAN in RSPAN session 2, to configure Gigabit Ethernet source port 2 as the destination interface, and to enable ingress forwarding on the interface with VLAN 6 as the default incoming VLAN.

```
Switch(config)# monitor session 2 source remote vlan 901
Switch(config)# monitor session 2 destination interface gigabitethernet0/2 ingress vlan 6
Switch(config)# end
```

Specifying VLANs to Filter

Beginning in privileged EXEC mode, follow these steps to configure the RSPAN source session to limit RSPAN source traffic to specific VLANs:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**no monitor session** {*session_number* \| **all** \| **local** \| **remote**}	Remove any existing SPAN configuration for the session. For *session_number*, the range is 1 to 66. Specify **all** to remove all SPAN sessions, **local** to remove all local sessions, or **remote** to remove all remote SPAN sessions.
Step 3	**monitor session** *session_number* **source interface** *interface-id*	Specify the characteristics of the source port (monitored port) and SPAN session. For *session_number*, the range is 1 to 66. For *interface-id*, specify the source port to monitor. The interface specified must already be configured as a trunk port.
Step 4	**monitor session** *session_number* **filter vlan** *vlan-id* [**,** \| **-**]	Limit the SPAN source traffic to specific VLANs. For *session_number*, enter the session number specified in step 3. For *vlan-id*, the range is 1 to 4094. (Optional) Use a comma (**,**) to specify a series of VLANs or use a hyphen (**-**) to specify a range of VLANs. Enter a space before and after the comma; enter a space before and after the hyphen.

	Command	Purpose
Step 5	**monitor session** *session_number* **destination remote vlan** *vlan-id*	Specify the RSPAN session and the destination remote VLAN (RSPAN VLAN). For *session_number*, enter the session number specified in step 3. For *vlan-id*, specify the RSPAN VLAN to carry the monitored traffic to the destination port.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show monitor** [**session** *session_number*] **show running-config**	Verify the configuration.
Step 8	**copy running-config startup-config**	(Optional) Save the configuration in the configuration file.

To monitor all VLANs on the trunk port, use the **no monitor session** *session_number* **filter vlan** global configuration command.

This example shows how to remove any existing configuration on RSPAN session 2, configure RSPAN session 2 to monitor traffic received on trunk port 2, and send traffic for only VLANs 2 through 5 and 9 to destination RSPAN VLAN 902.

```
Switch(config)# no monitor session 2
Switch(config)# monitor session 2 source interface gigabitethernet0/2 rx
Switch(config)# monitor session 2 filter vlan 2 - 5 , 9
Switch(config)# monitor session 2 destination remote vlan 902
Switch(config)# end
```

Displaying SPAN and RSPAN Status

To display the current SPAN or RSPAN configuration, use the **show monitor** user EXEC command. You can also use the **show running-config** privileged EXEC command to display configured SPAN or RSPAN sessions.



CHAPTER 25

Configuring RMON

This chapter describes how to configure Remote Network Monitoring (RMON) on the Cisco ME 3400 Ethernet Access switch.

RMON is a standard monitoring specification that defines a set of statistics and functions that can be exchanged between RMON-compliant console systems and network probes. RMON provides you with comprehensive network-fault diagnosis, planning, and performance-tuning information.

Note For complete syntax and usage information for the commands used in this chapter, see the "System Management Commands" section in the *Cisco IOS Configuration Fundamentals Command Reference, Release 12.2*.

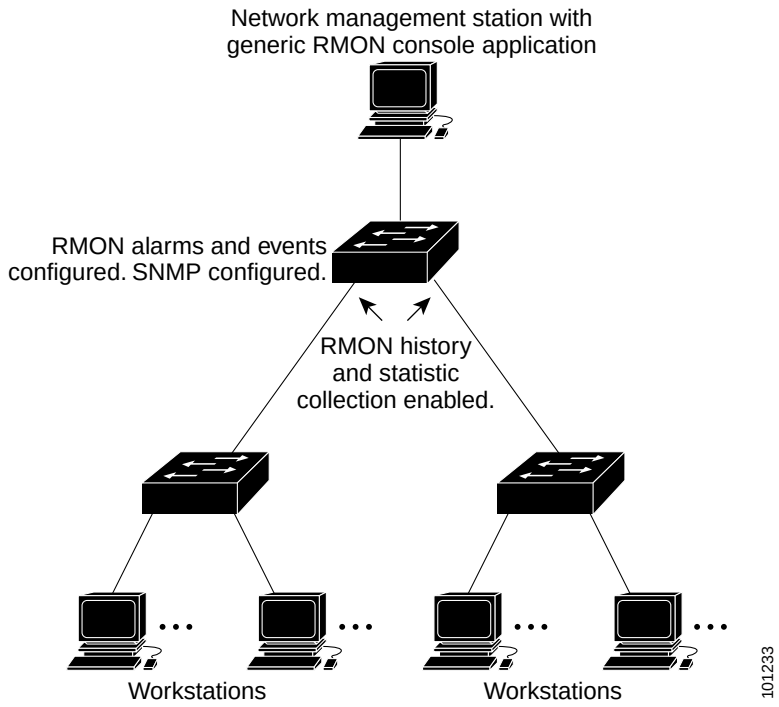
This chapter consists of these sections:

- Understanding RMON, page 25-1
- Configuring RMON, page 25-2
- Displaying RMON Status, page 25-6

Understanding RMON

RMON is an Internet Engineering Task Force (IETF) standard monitoring specification that allows various network agents and console systems to exchange network monitoring data. You can use the RMON feature with the Simple Network Management Protocol (SNMP) agent in the switch to monitor all the traffic flowing among switches on all connected LAN segments as shown in Figure 25-1.

Figure 25-1 Remote Monitoring Example



The switch supports these RMON groups (defined in RFC 1757):

- Statistics (RMON group 1)—Collects Ethernet statistics (including Fast Ethernet and Gigabit Ethernet statistics, depending on the switch type and supported interfaces) on an interface.
- History (RMON group 2)—Collects a history group of statistics on Ethernet ports (including Fast Ethernet and Gigabit Ethernet statistics, depending on the switch type and supported interfaces) for a specified polling interval.
- Alarm (RMON group 3)—Monitors a specific management information base (MIB) object for a specified interval, triggers an alarm at a specified value (rising threshold), and resets the alarm at another value (falling threshold). Alarms can be used with events; the alarm triggers an event, which can generate a log entry or an SNMP trap.
- Event (RMON group 9)—Specifies the action to take when an event is triggered by an alarm. The action can be to generate a log entry or an SNMP trap.

Because switches supported by this software release use hardware counters for RMON data processing, the monitoring is more efficient, and little processing power is required.

Configuring RMON

These sections contain this configuration information:

- Default RMON Configuration, page 25-3
- Configuring RMON Alarms and Events, page 25-3 (required)
- Collecting Group History Statistics on an Interface, page 25-5 (optional)
- Collecting Group Ethernet Statistics on an Interface, page 25-6 (optional)

Default RMON Configuration

RMON is disabled by default; no alarms or events are configured.

Only RMON 1 is supported on the switch.

Configuring RMON Alarms and Events

You can configure your switch for RMON by using the command-line interface (CLI) or an SNMP-compatible network management station. We recommend that you use a generic RMON console application on the network management station (NMS) to take advantage of the RMON network management capabilities. You must also configure SNMP on the switch to access RMON MIB objects. For more information, see Chapter 27, "Configuring SNMP."

Beginning in privileged EXEC mode, follow these steps to enable RMON alarms and events. This procedure is required.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**rmon alarm** *number variable interval* {**absolute** \| **delta**} **rising-threshold** *value* [*event-number*] **falling-threshold** *value* [*event-number*] [**owner** *string*]	Set an alarm on a MIB object. • For *number*, specify the alarm number. The range is 1 to 65535. • For *variable*, specify the MIB object to monitor. • For *interval*, specify the time in seconds the alarm monitors the MIB variable. The range is 1 to 4294967295 seconds. • Specify the **absolute** keyword to test each MIB variable directly. Specify the **delta** keyword to test the change between samples of a MIB variable. • For *value*, specify a number at which the alarm is triggered and one for when the alarm is reset. The range for the rising threshold and falling threshold values is -2147483648 to 2147483647. • (Optional) For *event-number*, specify the event number to trigger when the rising or falling threshold exceeds its limit. • (Optional) For **owner** *string*, specify the owner of the alarm.

	Command	Purpose
Step 3	**rmon event** *number* [**description** *string*] [**log**] [**owner** *string*] [**trap** *community*]	Add an event in the RMON event table that is associated with an RMON event number. • For *number*, assign an event number. The range is 1 to 65535. • (Optional) For **description** *string*, specify a description of the event. • (Optional) Use the **log** keyword to generate an RMON log entry when the event is triggered. • (Optional) For **owner** *string*, specify the owner of this event. • (Optional) For **trap** *community*, enter the SNMP community string used for this trap.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable an alarm, use the **no rmon alarm** *number* global configuration command on each alarm you configured. You cannot disable at once all the alarms that you configured. To disable an event, use the **no rmon event** *number* global configuration command. To learn more about alarms and events and how they interact with each other, see RFC 1757.

You can set an alarm on any MIB object. The following example configures RMON alarm number 10 by using the **rmon alarm** command. The alarm monitors the MIB variable *ifEntry.20.1* once every 20 seconds until the alarm is disabled and checks the change in the variable's rise or fall. If the *ifEntry.20.1* value shows a MIB counter increase of 15 or more, such as from 100000 to 100015, the alarm is triggered. The alarm in turn triggers event number 1, which is configured with the **rmon event** command. Possible events can include a log entry or an SNMP trap. If the *ifEntry.20.1* value changes by 0, the alarm is reset and can be triggered again.

```
Switch(config)# rmon alarm 10 ifEntry.20.1 20 delta rising-threshold 15 1
falling-threshold 0 owner jjohnson
```

The following example creates RMON event number 1 by using the **rmon event** command. The event is defined as *High ifOutErrors* and generates a log entry when the event is triggered by the alarm. The user *jjones* owns the row that is created in the event table by this command. This example also generates an SNMP trap when the event is triggered.

```
Switch(config)# rmon event 1 log trap eventtrap description "High ifOutErrors" owner
jjones
```

Collecting Group History Statistics on an Interface

You must first configure RMON alarms and events to display collection information.

Beginning in privileged EXEC mode, follow these steps to collect group history statistics on an interface. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface on which to collect history, and enter interface configuration mode.
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled, and NNIs are enabled.
Step 4	**rmon collection history** *index* [**buckets** *bucket-number*] [**interval** *seconds*] [**owner** *ownername*]	Enable history collection for the specified number of buckets and time period. • For *index*, identify the RMON group of statistics The range is 1 to 65535. • (Optional) For **buckets** *bucket-number,* specify the maximum number of buckets desired for the RMON collection history group of statistics. The range is 1 to 65535. The default is 50 buckets. • (Optional) For **interval** *seconds*, specify the number of seconds in each polling cycle. The range is 1 to 3600. The default is 1800 seconds. • (Optional) For **owner** *ownername*, enter the name of the owner of the RMON group of statistics.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config**	Verify your entries.
Step 7	**show rmon history**	Display the contents of the switch history table.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable history collection, use the **no rmon collection history** *index* interface configuration command.

Collecting Group Ethernet Statistics on an Interface

Beginning in privileged EXEC mode, follow these steps to collect group Ethernet statistics on an interface. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface on which to collect statistics, and enter interface configuration mode.
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled, and NNIs are enabled.
Step 4	**rmon collection stats** *index* [**owner** *ownername*]	Enable RMON statistic collection on the interface. • For *index*, specify the RMON group of statistics. The range is from 1 to 65535. • (Optional) For **owner** *ownername*, enter the name of the owner of the RMON group of statistics.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config**	Verify your entries.
Step 7	**show rmon statistics**	Display the contents of the switch statistics table.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable the collection of group Ethernet statistics, use the **no rmon collection stats** *index* interface configuration command.

This example shows how to collect RMON statistics for the owner *root*:

```
Switch(config)# interface fastethernet0/1
Switch(config)# no shutdown
Switch(config-if)# rmon collection stats 2 owner root
```

Displaying RMON Status

To display the RMON status, use one or more of the privileged EXEC commands in Table 25-1:

Table 25-1 Commands for Displaying RMON Status

Command	Purpose
show rmon	Displays general RMON statistics.
show rmon alarms	Displays the RMON alarm table.
show rmon events	Displays the RMON event table.
show rmon history	Displays the RMON history table.
show rmon statistics	Displays the RMON statistics table.

For information about the fields in these displays, see the "System Management Commands" section in the *Cisco IOS Configuration Fundamentals Command Reference, Release 12.2*.



CHAPTER 26

Configuring System Message Logging

This chapter describes how to configure system message logging on the Cisco ME 3400 Ethernet Access switch.

Note For complete syntax and usage information for the commands used in this chapter, see the *Cisco IOS Configuration Fundamentals Command Reference, Release 12.2*.

This chapter consists of these sections:

- Understanding System Message Logging, page 26-1
- Configuring System Message Logging, page 26-2
- Displaying the Logging Configuration, page 26-12

Understanding System Message Logging

By default, a switch sends the output from system messages and **debug** privileged EXEC commands to a logging process. The logging process controls the distribution of logging messages to various destinations, such as the logging buffer, terminal lines, or a UNIX syslog server, depending on your configuration. The process also sends messages to the console.

Note The syslog format is compatible with 4.3 BSD UNIX.

When the logging process is disabled, messages are sent only to the console. The messages are sent as they are generated, so message and debug output are interspersed with prompts or output from other commands. Messages appear on the console after the process that generated them has finished.

You can set the severity level of the messages to control the type of messages displayed on the consoles and each of the destinations. You can time-stamp log messages or set the syslog source address to enhance real-time debugging and management. For information on possible messages, see the system message guide for this release.

You can access logged system messages by using the switch command-line interface (CLI) or by saving them to a properly configured syslog server. The switch software saves syslog messages in an internal buffer.

You can remotely monitor system messages by viewing the logs on a syslog server or by accessing the switch through Telnet or through the console port.

Configuring System Message Logging

These sections contain this configuration information:

- System Log Message Format, page 26-2
- Default System Message Logging Configuration, page 26-3
- Disabling Message Logging, page 26-3 (optional)
- Setting the Message Display Destination Device, page 26-4 (optional)
- Synchronizing Log Messages, page 26-5 (optional)
- Enabling and Disabling Time Stamps on Log Messages, page 26-7 (optional)
- Enabling and Disabling Sequence Numbers in Log Messages, page 26-7 (optional)
- Defining the Message Severity Level, page 26-8 (optional)
- Limiting Syslog Messages Sent to the History Table and to SNMP, page 26-9 (optional)
- Configuring UNIX Syslog Servers, page 26-10 (optional)

System Log Message Format

System log messages can contain up to 80 characters and a percent sign (%), which follows the optional sequence number or time-stamp information, if configured. Messages appear in this format:

seq no:timestamp: %facility-severity-MNEMONIC:description

The part of the message preceding the percent sign depends on the setting of the **service sequence-numbers**, **service timestamps log datetime**, **service timestamps log datetime** [**localtime**] [**msec**] [**show-timezone**], or **service timestamps log uptime** global configuration command.

Table 26-1 describes the elements of syslog messages.

Table 26-1 System Log Message Elements

Element	Description
seq no:	Stamps log messages with a sequence number only if the **service sequence-numbers** global configuration command is configured. For more information, see the "Enabling and Disabling Sequence Numbers in Log Messages" section on page 26-7.
timestamp formats: *mm/dd hh:mm:ss* or *hh:mm:ss* (short uptime) or *d h* (long uptime)	Date and time of the message or event. This information appears only if the **service timestamps log** [**datetime** \| **log**] global configuration command is configured. For more information, see the "Enabling and Disabling Time Stamps on Log Messages" section on page 26-7.
facility	The facility to which the message refers (for example, SNMP, SYS, and so forth). For a list of supported facilities, see Table 26-4 on page 26-11.
severity	Single-digit code from 0 to 7 that is the severity of the message. For a description of the severity levels, see Table 26-3 on page 26-9.

Table 26-1 System Log Message Elements (continued)

Element	Description
MNEMONIC	Text string that uniquely describes the message.
description	Text string containing detailed information about the event being reported.

This example shows a partial switch system message:

```
00:00:46: %LINK-3-UPDOWN: Interface Port-channel1, changed state to up
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet0/1, changed state to up
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet0/2, changed state to up
00:00:48: %LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan1, changed state to down
00:00:48: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/1, changed
state to down 2
*Mar  1 18:46:11: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
18:47:02: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
*Mar  1 18:48:50.483 UTC: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
```

Default System Message Logging Configuration

Table 26-2 shows the default system message logging configuration.

Table 26-2 Default System Message Logging Configuration

Feature	Default Setting
System message logging to the console	Enabled.
Console severity	Debugging (and numerically lower levels; see Table 26-3 on page 26-9).
Logging file configuration	No filename specified.
Logging buffer size	4096 bytes.
Logging history size	1 message.
Time stamps	Disabled.
Synchronous logging	Disabled.
Logging server	Disabled.
Syslog server IP address	None configured.
Server facility	Local7 (see Table 26-4 on page 26-11).
Server severity	Informational (and numerically lower levels; see Table 26-3 on page 26-9).

Disabling Message Logging

Message logging is enabled by default. It must be enabled to send messages to any destination other than the console. When enabled, log messages are sent to a logging process, which logs messages to designated locations asynchronously to the processes that generated the messages.

Beginning in privileged EXEC mode, follow these steps to disable message logging. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**no logging console**	Disable message logging.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config** or **show logging**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Disabling the logging process can slow down the switch because a process must wait until the messages are written to the console before continuing. When the logging process is disabled, messages appear on the console as soon as they are produced, often appearing in the middle of command output.

The **logging synchronous** global configuration command also affects the display of messages to the console. When this command is enabled, messages appear only after you press Return. For more information, see the "Synchronizing Log Messages" section on page 26-5.

To re-enable message logging after it has been disabled, use the **logging on** global configuration command.

Setting the Message Display Destination Device

If message logging is enabled, you can send messages to specific locations in addition to the console. Beginning in privileged EXEC mode, use one or more of the following commands to specify the locations that receive messages. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**logging buffered** [*size*]	Log messages to an internal buffer on the switch. The default buffer size is 4096. The range is 4096 to 2147483647 bytes. If the switch fails, the log file is lost unless you previously saved it to Flash memory. See Step 4. **Note** Do not make the buffer size too large because the switch could run out of memory for other tasks. Use the **show memory** privileged EXEC command to view the free processor memory on the switch. However, this value is the maximum available, and the buffer size should *not* be set to this amount.

	Command	Purpose
Step 3	**logging** *host*	Log messages to a UNIX syslog server host. For *host*, specify the name or IP address of the host to be used as the syslog server. To build a list of syslog servers that receive logging messages, enter this command more than once. For complete syslog server configuration steps, see the "Configuring UNIX Syslog Servers" section on page 26-10.
Step 4	**logging file flash:***filename* [*max-file-size* [*min-file-size*]] [*severity-level-number* \| *type*]	Store log messages in a file in flash memory. • For *filename*, enter the log message filename. • (Optional) For *max-file-size*, specify the maximum logging file size. The range is 4096 to 2147483647. The default is 4096 bytes. • (Optional) For *min-file-size*, specify the minimum logging file size. The range is 1024 to 2147483647. The default is 2048 bytes. • (Optional) For *severity-level-number* \| *type*, specify either the logging severity level or the logging type. The severity range is 0 to 7. For a list of logging type keywords, see Table 26-3 on page 26-9. By default, the log file receives debugging messages and numerically lower levels.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**terminal monitor**	Log messages to a nonconsole terminal during the current session. Terminal parameter-setting commands are set locally and do not remain in effect after the session has ended. You must perform this step for each session to see the debugging messages.
Step 7	**show running-config**	Verify your entries.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

The **logging buffered** global configuration command copies logging messages to an internal buffer. The buffer is circular, so newer messages overwrite older messages after the buffer is full. To display the messages that are logged in the buffer, use the **show logging** privileged EXEC command. The first message displayed is the oldest message in the buffer. To clear the contents of the buffer, use the **clear logging** privileged EXEC command.

To disable logging to the console, use the **no logging console** global configuration command. To disable logging to a file, use the **no logging file** [*severity-level-number* | *type*] global configuration command.

Synchronizing Log Messages

You can synchronize unsolicited messages and **debug** privileged EXEC command output with solicited device output and prompts for a specific console port line or virtual terminal line. You can identify the types of messages to be output asynchronously based on the level of severity. You can also configure the maximum number of buffers for storing asynchronous messages for the terminal after which messages are dropped.

When synchronous logging of unsolicited messages and **debug** command output is enabled, unsolicited device output appears on the console or printed after solicited device output appears or is printed. Unsolicited messages and **debug** command output appears on the console after the prompt for user input

is returned. Therefore, unsolicited messages and **debug** command output are not interspersed with solicited device output and prompts. After the unsolicited messages appear, the console again displays the user prompt.

Beginning in privileged EXEC mode, follow these steps to configure synchronous logging. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**line** [**console** \| **vty**] *line-number* [*ending-line-number*]	Specify the line to be configured for synchronous logging of messages. • Use the **console** keyword for configurations that occur through the switch console port. • Use the **line vty** *line-number* command to specify which vty lines are to have synchronous logging enabled. You use a vty connection for configurations that occur through a Telnet session. The range of line numbers is from 0 to 15. You can change the setting of all 16 vty lines at once by entering: **line vty 0 15** Or you can change the setting of the single vty line being used for your current connection. For example, to change the setting for vty line 2, enter: **line vty 2** When you enter this command, the mode changes to line configuration.
Step 3	**logging synchronous** [**level** [*severity-level* \| **all**] \| **limit** *number-of-buffers*]	Enable synchronous logging of messages. • (Optional) For **level** *severity-level*, specify the message severity level. Messages with a severity level equal to or higher than this value are printed asynchronously. Low numbers mean greater severity and high numbers mean lesser severity. The default is 2. • (Optional) Specifying **level all** means that all messages are printed asynchronously regardless of the severity level. • (Optional) For **limit** *number-of-buffers*, specify the number of buffers to be queued for the terminal after which new messages are dropped. The range is 0 to 2147483647. The default is 20.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable synchronization of unsolicited messages and debug output, use the **no logging synchronous** [**level** *severity-level* | **all**] [**limit** *number-of-buffers*] line configuration command.

Enabling and Disabling Time Stamps on Log Messages

By default, log messages are not time-stamped.

Beginning in privileged EXEC mode, follow these steps to enable time-stamping of log messages. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**service timestamps log uptime** or **service timestamps log datetime** [**msec**] [**localtime**] [**show-timezone**]	Enable log time stamps. The first command enables time stamps on log messages, showing the time since the system was rebooted. The second command enables time stamps on log messages. Depending on the options selected, the time stamp can include the date, time in milliseconds relative to the local time-zone, and the time zone name.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable time stamps for both debug and log messages, use the **no service timestamps** global configuration command.

This example shows part of a logging display with the **service timestamps log datetime** global configuration command enabled:

```
*Mar  1 18:46:11: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
```

This example shows part of a logging display with the s**ervice timestamps log uptime** global configuration command enabled:

```
00:00:46: %LINK-3-UPDOWN: Interface Port-channel1, changed state to up
```

Enabling and Disabling Sequence Numbers in Log Messages

Because there is a chance that more than one log message can have the same time stamp, you can display messages with sequence numbers so that you can unambiguously see a single message. By default, sequence numbers in log messages are not displayed.

Beginning in privileged EXEC mode, follow these steps to enable sequence numbers in log messages. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**service sequence-numbers**	Enable sequence numbers.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable sequence numbers, use the **no service sequence-numbers** global configuration command.

This example shows part of a logging display with sequence numbers enabled:

```
000019: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
```

Defining the Message Severity Level

You can limit messages displayed to the selected device by specifying the severity level of the message, which are described in Table 26-3.

Beginning in privileged EXEC mode, follow these steps to define the message severity level. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**logging console** *level*	Limit messages logged to the console. By default, the console receives debugging messages and numerically lower levels (see Table 26-3 on page 26-9).
Step 3	**logging monitor** *level*	Limit messages logged to the terminal lines. By default, the terminal receives debugging messages and numerically lower levels (see Table 26-3 on page 26-9).
Step 4	**logging trap** *level*	Limit messages logged to the syslog servers. By default, syslog servers receive informational messages and numerically lower levels (see Table 26-3 on page 26-9). For complete syslog server configuration steps, see the "Configuring UNIX Syslog Servers" section on page 26-10.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config** or **show logging**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Note Specifying a *level* causes messages at that level and numerically lower levels to appear at the destination.

To disable logging to the console, use the **no logging console** global configuration command. To disable logging to a terminal other than the console, use the **no logging monitor** global configuration command. To disable logging to syslog servers, use the **no logging trap** global configuration command.

Table 26-3 describes the *level* keywords. It also lists the corresponding UNIX syslog definitions from the most severe level to the least severe level.

Table 26-3 Message Logging Level Keywords

Level Keyword	Level	Description	Syslog Definition
emergencies	0	System unstable	LOG_EMERG
alerts	1	Immediate action needed	LOG_ALERT
critical	2	Critical conditions	LOG_CRIT
errors	3	Error conditions	LOG_ERR
warnings	4	Warning conditions	LOG_WARNING
notifications	5	Normal but significant condition	LOG_NOTICE
informational	6	Informational messages only	LOG_INFO
debugging	7	Debugging messages	LOG_DEBUG

The software generates four other categories of messages:

- Error messages about software or hardware malfunctions, displayed at levels **warnings** through **emergencies**. These types of messages mean that the functionality of the switch is affected. For information on how to recover from these malfunctions, see the system message guide for this release.
- Output from the **debug** commands, displayed at the **debugging** level. Debug commands are typically used only by the Technical Assistance Center.
- Interface up or down transitions and system restart messages, displayed at the **notifications** level. This message is only for information; switch functionality is not affected.

Limiting Syslog Messages Sent to the History Table and to SNMP

If you enabled syslog message traps to be sent to an SNMP network management station by using the **snmp-server enable trap** global configuration command, you can change the level of messages sent and stored in the switch history table. You also can change the number of messages that are stored in the history table.

Messages are stored in the history table because SNMP traps are not guaranteed to reach their destination. By default, one message of the level **warning** and numerically lower levels (see Table 26-3 on page 26-9) are stored in the history table even if syslog traps are not enabled.

Beginning in privileged EXEC mode, follow these steps to change the level and history table size defaults. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**logging history** *level*[1]	Change the default level of syslog messages stored in the history file and sent to the SNMP server. See Table 26-3 on page 26-9 for a list of *level* keywords. By default, **warnings**, **errors**, **critical**, **alerts**, and **emergencies** messages are sent.

	Command	Purpose
Step 3	**logging history size** *number*	Specify the number of syslog messages that can be stored in the history table. The default is to store one message. The range is 0 to 500 messages.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

1. Table 26-3 lists the level keywords and severity level. For SNMP usage, the severity level values increase by 1. For example, emergencies equal 1, not 0, and critical equals 3, not 2.

When the history table is full (it contains the maximum number of message entries specified with the **logging history size** global configuration command), the oldest message entry is deleted from the table to allow the new message entry to be stored.

To return the logging of syslog messages to the default level, use the **no logging history** global configuration command. To return the number of messages in the history table to the default value, use the **no logging history size** global configuration command.

Configuring UNIX Syslog Servers

The next sections describe how to configure the UNIX server syslog daemon and how to define the UNIX system logging facility.

Logging Messages to a UNIX Syslog Daemon

Before you can send system log messages to a UNIX syslog server, you must configure the syslog daemon on a UNIX server. This procedure is optional.

Log in as root, and perform these steps:

Note Some recent versions of UNIX syslog daemons no longer accept by default syslog packets from the network. If this is the case with your system, use the UNIX **man syslogd** command to decide what options must be added to or removed from the syslog command line to enable logging of remote syslog messages.

Step 1 Add a line such as the following to the file /etc/syslog.conf:

`local7.debug /usr/adm/logs/`*`cisco.log`*

The **local7** keyword specifies the logging facility to be used; see Table 26-4 on page 26-11 for information on the facilities. The **debug** keyword specifies the syslog level; see Table 26-3 on page 26-9 for information on the severity levels. The syslog daemon sends messages at this level or at a more severe level to the file specified in the next field. The file must already exist, and the syslog daemon must have permission to write to it.

Step 2 Create the log file by entering these commands at the UNIX shell prompt:

```
$ touch /var/log/cisco.log
$ chmod 666 /var/log/cisco.log
```

Step 3 Make sure the syslog daemon reads the new changes:

```
$ kill -HUP `cat /etc/syslog.pid`
```

For more information, see the **man syslog.conf** and **man syslogd** commands on your UNIX system.

Configuring the UNIX System Logging Facility

When sending system log messages to an external device, you can cause the switch to identify its messages as originating from any of the UNIX syslog facilities.

Beginning in privileged EXEC mode, follow these steps to configure UNIX system facility message logging. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**logging** *host*	Log messages to a UNIX syslog server host by entering its IP address. To build a list of syslog servers that receive logging messages, enter this command more than once.
Step 3	**logging trap** *level*	Limit messages logged to the syslog servers. Be default, syslog servers receive informational messages and lower. See Table 26-3 on page 26-9 for *level* keywords.
Step 4	**logging facility** *facility-type*	Configure the syslog facility. See Table 26-4 on page 26-11 for *facility-type* keywords. The default is **local7**.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove a syslog server, use the **no logging** *host* global configuration command, and specify the syslog server IP address. To disable logging to syslog servers, enter the **no logging trap** global configuration command.

Table 26-4 lists the UNIX system facilities supported by the software. For more information about these facilities, consult the operator's manual for your UNIX operating system.

Table 26-4 Logging Facility-Type Keywords

Facility Type Keyword	Description
auth	Authorization system
cron	Cron facility
daemon	System daemon
kern	Kernel
local0-7	Locally defined messages
lpr	Line printer system

Table 26-4 Logging Facility-Type Keywords (continued)

Facility Type Keyword	Description
mail	Mail system
news	USENET news
sys9-14	System use
syslog	System log
user	User process
uucp	UNIX-to-UNIX copy system

Displaying the Logging Configuration

To display the logging configuration and the contents of the log buffer, use the **show logging** privileged EXEC command. For information about the fields in this display, see the *Cisco IOS Configuration Fundamentals Command Reference, Release 12.2*.



Configuring SNMP

This chapter describes how to configure the Simple Network Management Protocol (SNMP) on the Cisco ME 3400 Ethernet Access switch.

Note For complete syntax and usage information for the commands used in this chapter, see the switch command reference for this release and to the *Cisco IOS Configuration Fundamentals Command Reference, Release 12.2*.

This chapter consists of these sections:

- Understanding SNMP, page 27-1
- Configuring SNMP, page 27-6
- Displaying SNMP Status, page 27-16

Understanding SNMP

SNMP is an application-layer protocol that provides a message format for communication between managers and agents. The SNMP system consists of an SNMP manager, an SNMP agent, and a MIB. The SNMP manager can be part of a network management system (NMS) such as CiscoWorks. The agent and MIB reside on the switch. To configure SNMP on the switch, you define the relationship between the manager and the agent.

The SNMP agent contains MIB variables whose values the SNMP manager can request or change. A manager can get a value from an agent or store a value into the agent. The agent gathers data from the MIB, the repository for information about device parameters and network data. The agent can also respond to a manager's requests to get or set data.

An agent can send unsolicited traps to the manager. Traps are messages alerting the SNMP manager to a condition on the network. Traps can mean improper user authentication, restarts, link status (up or down), MAC address tracking, closing of a TCP connection, loss of connection to a neighbor, or other significant events.

These sections contain this conceptual information:

- SNMP Versions, page 27-2
- SNMP Manager Functions, page 27-3
- SNMP Agent Functions, page 27-4
- SNMP Community Strings, page 27-4

- Using SNMP to Access MIB Variables, page 27-4
- SNMP Notifications, page 27-5
- SNMP ifIndex MIB Object Values, page 27-5

SNMP Versions

This software release supports these SNMP versions:

- SNMPv1—The Simple Network Management Protocol, a Full Internet Standard, defined in RFC 1157.
- SNMPv2C replaces the Party-based Administrative and Security Framework of SNMPv2Classic with the community-string-based Administrative Framework of SNMPv2C while retaining the bulk retrieval and improved error handling of SNMPv2Classic. It has these features:
 - SNMPv2—Version 2 of the Simple Network Management Protocol, a Draft Internet Standard, defined in RFCs 1902 through 1907.
 - SNMPv2C—The community-string-based Administrative Framework for SNMPv2, an Experimental Internet Protocol defined in RFC 1901.
- SNMPv3—Version 3 of the SNMP is an interoperable standards-based protocol defined in RFCs 2273 to 2275. SNMPv3 provides secure access to devices by authenticating and encrypting packets over the network and includes these security features:
 - Message integrity—ensuring that a packet was not tampered with in transit
 - Authentication—determining that the message is from a valid source
 - Encryption—mixing the contents of a package to prevent it from being read by an unauthorized source.

Note To select encryption, enter the **priv** keyword. This keyword is available only when the cryptographic (encrypted) software image is installed.

Both SNMPv1 and SNMPv2C use a community-based form of security. The community of managers able to access the agent's MIB is defined by an IP address access control list and password.

SNMPv2C includes a bulk retrieval mechanism and more detailed error message reporting to management stations. The bulk retrieval mechanism retrieves tables and large quantities of information, minimizing the number of round-trips required. The SNMPv2C improved error-handling includes expanded error codes that distinguish different kinds of error conditions; these conditions are reported through a single error code in SNMPv1. Error return codes in SNMPv2C report the error type.

SNMPv3 provides for both security models and security levels. A security model is an authentication strategy set up for a user and the group within which the user resides. A security level is the permitted level of security within a security model. A combination of the security level and the security model determine which security mechanism is used when handling an SNMP packet. Available security models are SNMPv1, SNMPv2C, and SNMPv3.

Table 27-1 identifies the characteristics of the different combinations of security models and levels.

Table 27-1 SNMP Security Models and Levels

Model	Level	Authentication	Encryption	Result
SNMPv1	noAuthNoPriv	Community string	No	Uses a community string match for authentication.
SNMPv2C	noAuthNoPriv	Community string	No	Uses a community string match for authentication.
SNMPv3	noAuthNoPriv	Username	No	Uses a username match for authentication.
SNMPv3	authNoPriv	MD5 or SHA	No	Provides authentication based on the HMAC-MD5 or HMAC-SHA algorithms.
SNMPv3	authPriv (requires the cryptographic software image)	MD5 or SHA	DES	Provides authentication based on the HMAC-MD5 or HMAC-SHA algorithms. Provides DES 56-bit encryption in addition to authentication based on the CBC-DES (DES-56) standard.

You must configure the SNMP agent to use the SNMP version supported by the management station. Because an agent can communicate with multiple managers, you can configure the software to support communications using SNMPv1, SNMPv2C, or SNMPv3.

SNMP Manager Functions

The SNMP manager uses information in the MIB to perform the operations described in Table 27-2.

Table 27-2 SNMP Operations

Operation	Description
get-request	Retrieves a value from a specific variable.
get-next-request	Retrieves a value from a variable within a table.[1]
get-bulk-request[2]	Retrieves large blocks of data, such as multiple rows in a table, that would otherwise require the transmission of many small blocks of data.
get-response	Replies to a get-request, get-next-request, and set-request sent by an NMS.
set-request	Stores a value in a specific variable.
trap	An unsolicited message sent by an SNMP agent to an SNMP manager when some event has occurred.

1. With this operation, an SNMP manager does not need to know the exact variable name. A sequential search is performed to find the needed variable from within a table.
2. The **get-bulk** command only works with SNMPv2 or later.

SNMP Agent Functions

The SNMP agent responds to SNMP manager requests as follows:

- Get a MIB variable—The SNMP agent begins this function in response to a request from the NMS. The agent retrieves the value of the requested MIB variable and responds to the NMS with that value.
- Set a MIB variable—The SNMP agent begins this function in response to a message from the NMS. The SNMP agent changes the value of the MIB variable to the value requested by the NMS.

The SNMP agent also sends unsolicited trap messages to notify an NMS that a significant event has occurred on the agent. Examples of trap conditions include, but are not limited to, when a port or module goes up or down, when spanning-tree topology changes occur, and when authentication failures occur.

SNMP Community Strings

SNMP community strings authenticate access to MIB objects and function as embedded passwords. In order for the NMS to access the switch, the community string definitions on the NMS must match at least one of the three community string definitions on the switch.

A community string can have one of these attributes:

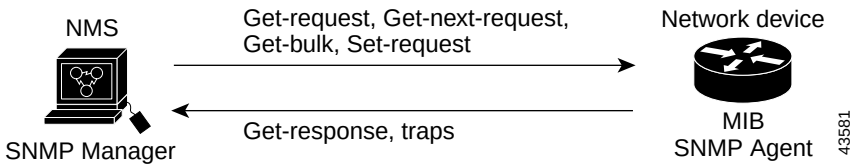
- Read-only (RO)—Gives read access to authorized management stations to all objects in the MIB except the community strings, but does not allow write access
- Read-write (RW)—Gives read and write access to authorized management stations to all objects in the MIB, but does not allow access to the community strings

Using SNMP to Access MIB Variables

An example of an NMS is the CiscoWorks network management software. CiscoWorks 2000 software uses the switch MIB variables to set device variables and to poll devices on the network for specific information. The results of a poll can be displayed as a graph and analyzed to troubleshoot internetworking problems, increase network performance, verify the configuration of devices, monitor traffic loads, and more.

As shown in Figure 27-1, the SNMP agent gathers data from the MIB. The agent can send traps, or notification of certain events, to the SNMP manager, which receives and processes the traps. Traps alert the SNMP manager to a condition on the network such as improper user authentication, restarts, link status (up or down), MAC address tracking, and so forth. The SNMP agent also responds to MIB-related queries sent by the SNMP manager in *get-request*, *get-next-request*, and *set-request* format.

Figure 27-1 SNMP Network



For information on supported MIBs and how to access them, see Appendix A, “Supported MIBs.”

SNMP Notifications

SNMP allows the switch to send notifications to SNMP managers when particular events occur. SNMP notifications can be sent as traps or inform requests. In command syntax, unless there is an option in the command to select either traps or informs, the keyword *traps* refers to either traps or informs, or both. Use the **snmp-server host** command to specify whether to send SNMP notifications as traps or informs.

Note SNMPv1 does not support informs.

Traps are unreliable because the receiver does not send an acknowledgment when it receives a trap, and the sender cannot determine if the trap was received. When an SNMP manager receives an inform request, it acknowledges the message with an SNMP response protocol data unit (PDU). If the sender does not receive a response, the inform request can be sent again. Because they can be re-sent, informs are more likely than traps to reach their intended destination.

The characteristics that make informs more reliable than traps also consume more resources in the switch and in the network. Unlike a trap, which is discarded as soon as it is sent, an inform request is held in memory until a response is received or the request times out. Traps are sent only once, but an inform might be re-sent or retried several times. The retries increase traffic and contribute to a higher overhead on the network. Therefore, traps and informs require a trade-off between reliability and resources. If it is important that the SNMP manager receive every notification, use inform requests. If traffic on the network or memory in the switch is a concern and notification is not required, use traps.

SNMP ifIndex MIB Object Values

In an NMS, the IF-MIB generates and assigns an interface index (ifIndex) object value that is a unique number greater than zero to identify a physical or a logical interface. When the switch reboots or the switch software is upgraded, the switch uses this same value for the interface. For example, if the switch assigns a port 2 an ifIndex value of 10003, this value is the same after the switch reboots.

The switch uses one of the values in Table 27-3 to assign an ifIndex value to an interface:

Table 27-3 ifIndex Values

Interface Type	ifIndex Range
SVI[1]	1–4999
EtherChannel	5000–5012
Loopback	5013–5077
Tunnel	5078–5142
Physical (such as Gigabit Ethernet or SFP[2]-module interfaces)	10000–14500
Null	14501

1. SVI = switch virtual interface
2. SFP = small form-factor pluggable

Note The switch might not use sequential values within a range.

Configuring SNMP

These sections contain this configuration information:

- Default SNMP Configuration, page 27-6
- SNMP Configuration Guidelines, page 27-6
- Disabling the SNMP Agent, page 27-7
- Configuring Community Strings, page 27-8
- Configuring SNMP Groups and Users, page 27-9
- Configuring SNMP Notifications, page 27-11
- Setting the Agent Contact and Location Information, page 27-14
- Limiting TFTP Servers Used Through SNMP, page 27-15
- SNMP Examples, page 27-15

Default SNMP Configuration

Table 27-4 shows the default SNMP configuration.

Table 27-4 Default SNMP Configuration

Feature	Default Setting
SNMP agent	Disabled[1].
SNMP trap receiver	None configured.
SNMP traps	None enabled except the trap for TCP connections (**tty**).
SNMP version	If no **version** keyword is present, the default is Version 1.
SNMPv3 authentication	If no keyword is entered, the default is the **noauth** (noAuthNoPriv) security level.
SNMP notification type	If no type is specified, all notifications are sent.

1. This is the default when the switch starts and the startup configuration does not have any **snmp-server** global configuration commands.

SNMP Configuration Guidelines

If the switch starts and the witch startup configuration has at least one **snmp-server** global configuration command, the SNMP agent is enabled.

An SNMP *group* is a table that maps SNMP users to SNMP views. An SNMP *user* is a member of an SNMP group. An SNMP *host* is the recipient of an SNMP trap operation. An SNMP *engine ID* is a name for the local or remote SNMP engine.

When configuring SNMP, follow these guidelines:

- When configuring an SNMP group, do not specify a notify view. The **snmp-server host** global configuration command autogenerates a notify view for the user and then adds it to the group associated with that user. Modifying the group's notify view affects all users associated with that group. See the *Cisco IOS Configuration Fundamentals Command Reference, Release 12.2* for information about when you should configure notify views.
- To configure a remote user, specify the IP address or port number for the remote SNMP agent of the device where the user resides.
- Before you configure remote users for a particular agent, configure the SNMP engine ID, using the **snmp-server engineID** global configuration with the **remote** option. The remote agent's SNMP engine ID and user password are used to compute the authentication and privacy digests. If you do not configure the remote engine ID first, the configuration command fails.
- When configuring SNMP informs, you need to configure the SNMP engine ID for the remote agent in the SNMP database before you can send proxy requests or informs to it.
- If a local user is not associated with a remote host, the switch does not send informs for the **auth** (authNoPriv) and the **priv** (authPriv) authentication levels.
- Changing the value of the SNMP engine ID has important side effects. A user's password (entered on the command line) is converted to an MD5 or SHA security digest based on the password and the local engine ID. The command-line password is then destroyed, as required by RFC 2274. Because of this deletion, if the value of the engine ID changes, the security digests of SNMPv3 users become invalid, and you need to reconfigure SNMP users by using the **snmp-server user** *username* global configuration command. Similar restrictions require the reconfiguration of community strings when the engine ID changes.

Disabling the SNMP Agent

Beginning in privileged EXEC mode, follow these steps to disable the SNMP agent:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**no snmp-server**	Disable the SNMP agent operation.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

The **no snmp-server** global configuration command disables all running versions (Version 1, Version 2C, and Version 3) on the device. No specific Cisco IOS command exists to enable SNMP. The first **snmp-server** global configuration command that you enter enables all versions of SNMP.

Configuring Community Strings

You use the SNMP community string to define the relationship between the SNMP manager and the agent. The community string acts like a password to permit access to the agent on the switch. Optionally, you can specify one or more of these characteristics associated with the string:

- An access list of IP addresses of the SNMP managers that are permitted to use the community string to gain access to the agent
- A MIB view, which defines the subset of all MIB objects accessible to the given community
- Read and write or read-only permission for the MIB objects accessible to the community

Beginning in privileged EXEC mode, follow these steps to configure a community string on the switch:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**snmp-server community** *string* [**view** *view-name*] [**ro** \| **rw**] [*access-list-name or number*]	Configure the community string. • For *string*, specify a string that acts like a password and permits access to the SNMP protocol. You can configure one or more community strings of any length. • (Optional) For **view**, specify the view record accessible to the community. • (Optional) Specify either read-only (**ro**) if you want authorized management stations to retrieve MIB objects, or specify read-write (**rw**) if you want authorized management stations to retrieve and modify MIB objects. By default, the community string permits read-only access to all objects. • (Optional) For *access-list-number*, enter an IP standard access list numbered from 1 to 99 and 1300 to 1999.
Step 3	**access-list** *access-list-number* {**deny** \| **permit**} *source* [*source-wildcard*]	(Optional) If you specified an IP standard access list number in Step 2, then create the list, repeating the command as many times as necessary. • For *access-list-number*, enter the access list number specified in Step 2. • The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. • For *source*, enter the IP address of the SNMP managers that are permitted to use the community string to gain access to the agent. • (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
Step 4	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.



Note To disable access for an SNMP community, set the community string for that community to the null string (do not enter a value for the community string).

To remove a specific community string, use the **no snmp-server community** *string* global configuration command.

This example shows how to assign the string *comaccess* to SNMP, to allow read-only access, and to specify that IP access list 4 can use the community string to gain access to the switch SNMP agent:

```
Switch(config)# snmp-server community comaccess ro 4
```

Configuring SNMP Groups and Users

You can specify an identification name (engine ID) for the local or remote SNMP server engine on the switch. You can configure an SNMP server group that maps SNMP users to SNMP views, and you can add new users to the SNMP group.

Beginning in privileged EXEC mode, follow these steps to configure SNMP on the switch:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**snmp-server engineID** {**local** *engineid-string* \| **remote** *ip-address* [**udp-port** *port-number*] *engineid-string*}	Configure a name for either the local or remote copy of SNMP. • The *engineid-string* is a 24-character ID string with the name of the copy of SNMP. You need not specify the entire 24-character engine ID if it has trailing zeros. Specify only the portion of the engine ID up to the point where only zeros remain in the value. For example, to configure an engine ID of 123400000000000000000000, you can enter this: **snmp-server engineID local 1234** • If you select **remote**, specify the *ip-address* of the device that contains the remote copy of SNMP and the optional User Datagram Protocol (UDP) port to use for storing data on the remote device. The default is 162.

	Command	Purpose
Step 3	**snmp-server group** *groupname* {**v1** \| **v2c** \| **v3** {**auth** \| **noauth** \| **priv**}} [**read** *readview*] [**write** *writeview*] [**notify** *notifyview*] [**access** *access-list*]	Configure a new SNMP group on the remote device. • For *groupname*, specify the name of the group. • Specify a security model: – **v1** is the least secure of the possible security models. – **v2c** is the second least secure model. It allows transmission of informs and integers twice the normal width. – **v3,** the most secure, requires you to select an authentication level: **auth**—Enables the Message Digest 5 (MD5) and the Secure Hash Algorithm (SHA) packet authentication. **noauth**—Enables the noAuthNoPriv security level. This is the default if no keyword is specified. **priv**—Enables Data Encryption Standard (DES) packet encryption (also called *privacy*). **Note** The **priv** keyword is available only when the cryptographic software image is installed. • (Optional) Enter **read** *readview* with a string (not to exceed 64 characters) that is the name of the view in which you can only view the contents of the agent. • (Optional) Enter **write** *writeview* with a string (not to exceed 64 characters) that is the name of the view in which you enter data and configure the contents of the agent. • (Optional) Enter **notify** *notifyview* with a string (not to exceed 64 characters) that is the name of the view in which you specify a notify, inform, or trap. • (Optional) Enter **access** *access-list* with a string (not to exceed 64 characters) that is the name of the access list.

	Command	Purpose
Step 4	**snmp-server user** *username groupname* {**remote** *host* [**udp-port** *port*]} {**v1** [**access** *access-list*] \| **v2c** [**access** *access-list*] \| **v3** [**encrypted**] [**access** *access-list*] [**auth** {**md5** \| **sha**} *auth-password*]}	Add a new user for an SNMP group. • The *username* is the name of the user on the host that connects to the agent. • The *groupname* is the name of the group to which the user is associated. • Enter **remote** to specify a remote SNMP entity to which the user belongs and the hostname or IP address of that entity with the optional UDP port number. The default is 162. • Enter the SNMP version number (**v1**, **v2c**, or **v3**). If you enter **v3**, you have these additional options: – **encrypted** specifies that the password appears in encrypted format. This keyword is available only when the **v3** keyword is specified. – **auth** is an authentication level setting session that can be either the HMAC-MD5-96 (**md5**) or the HMAC-SHA-96 (**sha**) authentication level and requires a password string (not to exceed 64 characters). • (Optional) Enter **access** *access-list* with a string (not to exceed 64 characters) that is the name of the access list.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Configuring SNMP Notifications

A trap manager is a management station that receives and processes traps. Traps are system alerts that the switch generates when certain events occur. By default, no trap manager is defined, and no traps are sent. Switches running this Cisco IOS release can have an unlimited number of trap managers.

Note Many commands use the word *traps* in the command syntax. Unless there is an option in the command to select either traps or informs, the keyword *traps* refers to either traps, informs, or both. Use the **snmp-server host** global configuration command to specify whether to send SNMP notifications as traps or informs.

Table 27-5 describes the supported switch traps (notification types). You can enable any or all of these traps and configure a trap manager to receive them.

Table 27-5 Switch Notification Types

Notification Type Keyword	Description
bgp	Generates Border Gateway Protocol (BGP) state change traps. This option is only available when the metro IP access image is installed.
bridge	Generates STP bridge MIB traps.
config	Generates a trap for SNMP configuration changes.
config-copy	Generates a trap for SNMP copy configuration changes.
entity	Generates a trap for SNMP entity changes.
envmon	Generates environmental monitor traps. You can enable any or all of these environmental traps: fan, shutdown, status, supply, temperature.
flash	Generates SNMP FLASH notifications.
hsrp	Generates a trap for Hot Standby Router Protocol (HSRP) changes.
ipmulticast	Generates a trap for IP multicast routing changes.
mac-notification	Generates a trap for MAC address notifications.
msdp	Generates a trap for Multicast Source Discovery Protocol (MSDP) changes.
ospf	Generates a trap for Open Shortest Path First (OSPF) changes. You can enable any or all of these traps: Cisco specific, errors, link-state advertisement, rate limit, retransmit, and state changes.
pim	Generates a trap for Protocol-Independent Multicast (PIM) changes. You can enable any or all of these traps: invalid PIM messages, neighbor changes, and rendezvous point (RP)-mapping changes.
port-security	Generates SNMP port security traps. You can also set a maximum trap rate per second. The range is from 0 to 1000; the default is 0, which means that there is no rate limit.
rtr	Generates a trap for the SNMP Response Time Reporter (RTR).
snmp	Generates a trap for SNMP-type notifications for authentication, cold start, warm start, link up or link down.
storm-control	Generates a trap for SNMP storm-control. You can also set a maximum trap rate per second. The range is from 0 to 1000; the default is 0 (no limit is imposed; a trap is sent at every occurrence).
stpx	Generates SNMP STP Extended MIB traps.
syslog	Generates SNMP syslog traps.
tty	Generates a trap for TCP connections. This trap is enabled by default.
vlan-membership	Generates a trap for SNMP VLAN membership changes.
vlancreate	Generates SNMP VLAN created traps.
vlandelete	Generates SNMP VLAN deleted traps.

Note Though visible in the command-line help strings, the **cpu** [**threshold**], **flash insertion, flash removal, fru-ctrl**, and **vtp** keywords are not supported. The **snmp-server enable informs** global configuration command is not supported. To enable the sending of SNMP inform notifications, use the **snmp-server enable traps** global configuration command combined with the **snmp-server host** *host-addr* **informs** global configuration command.

You can use the **snmp-server host** global configuration command to a specific host to receive the notification types listed in Table 27-5.

Beginning in privileged EXEC mode, follow these steps to configure the switch to send traps or informs to a host:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**snmp-server engineID remote** *ip-address engineid-string*	Specify the engine ID for the remote host.
Step 3	**snmp-server user** *username groupname* {**remote** *host* [**udp-port** *port*]} {**v1** [**access** *access-list*] \| **v2c** [**access** *access-list*] \| **v3** [**encrypted**] [**access** *access-list*] [**auth** {**md5** \| **sha**} *auth-password*]}	Configure an SNMP user to be associated with the remote host created in Step 2. **Note** You cannot configure a remote user for an address without first configuring the engine ID for the remote host. Otherwise, you receive an error message, and the command is not executed.
Step 4	**snmp-server group** *groupname* {**v1** \| **v2c** \| **v3** {**auth** \| **noauth** \| **priv**}} [**read** *readview*] [**write** *writeview*] [**notify** *notifyview*] [**access** *access-list*]	Configure an SNMP group.
Step 5	**snmp-server host** *host-addr* [**informs** \| **traps**] [**version** {**1** \| **2c** \| **3** {**auth** \| **noauth** \| **priv**}}] *community-string* [*notification-type*]	Specify the recipient of an SNMP trap operation. • For *host-addr*, specify the name or Internet address of the host (the targeted recipient). • (Optional) Enter **informs** to send SNMP informs to the host. • (Optional) Enter **traps** (the default) to send SNMP traps to the host. • (Optional) Specify the SNMP **version** (**1**, **2c**, or **3**). SNMPv1 does not support informs. • (Optional) For Version 3, select authentication level **auth, noauth**, or **priv**. **Note** The **priv** keyword is available only when the cryptographic software image is installed. • For *community-string*, when **version 1** or **version 2c** is specified, enter the password-like community string sent with the notification operation. When **version 3** is specified, enter the SNMPv3 username. • (Optional) For *notification-type*, use the keywords listed in Table 27-5 on page 27-12. If no type is specified, all notifications are sent.

	Command	Purpose
Step 6	**snmp-server enable traps** *notification-types*	Enable the switch to send traps or informs and specify the type of notifications to be sent. For a list of notification types, see Table 27-5 on page 27-12, or enter **snmp-server enable traps ?** To enable multiple types of traps, you must enter a separate **snmp-server enable traps** command for each trap type.
Step 7	**snmp-server trap-source** *interface-id*	(Optional) Specify the source interface, which provides the IP address for the trap message. This command also sets the source IP address for informs.
Step 8	**snmp-server queue-length** *length*	(Optional) Establish the message queue length for each trap host. The range is 1 to 1000; the default is 10.
Step 9	**snmp-server trap-timeout** *seconds*	(Optional) Define how often to resend trap messages. The range is 1 to 1000; the default is 30 seconds.
Step 10	**end**	Return to privileged EXEC mode.
Step 11	**show running-config**	Verify your entries.
Step 12	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

The **snmp-server host** command specifies which hosts receive the notifications. The **snmp-server enable trap** command globally enables the mechanism for the specified notification (for traps and informs). To enable a host to receive an inform, you must configure an **snmp-server host informs** command for the host and globally enable informs by using the **snmp-server enable traps** command.

To remove the specified host from receiving traps, use the **no snmp-server host** *host* global configuration command. The **no snmp-server host** command with no keywords disables traps, but not informs, to the host. To disable informs, use the **no snmp-server host informs** global configuration command. To disable a specific trap type, use the **no snmp-server enable traps** *notification-types* global configuration command.

Setting the Agent Contact and Location Information

Beginning in privileged EXEC mode, follow these steps to set the system contact and location of the SNMP agent so that these descriptions can be accessed through the configuration file:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**snmp-server contact** *text*	Set the system contact string. For example: `snmp-server contact Dial System Operator at beeper 21555.`
Step 3	**snmp-server location** *text*	Set the system location string. For example: `snmp-server location Building 3/Room 222`
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Limiting TFTP Servers Used Through SNMP

Beginning in privileged EXEC mode, follow these steps to limit the TFTP servers used for saving and loading configuration files through SNMP to the servers specified in an access list:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**snmp-server tftp-server-list** *access-list-number*	Limit TFTP servers used for configuration file copies through SNMP to the servers in the access list. For *access-list-number*, enter an IP standard access list numbered from 1 to 99 and 1300 to 1999.
Step 3	**access-list** *access-list-number* {**deny** \| **permit**} *source* [*source-wildcard*]	Create a standard access list, repeating the command as many times as necessary. • For *access-list-number*, enter the access list number specified in Step 2. • The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. • For *source*, enter the IP address of the TFTP servers that can access the switch. • (Optional) For *source-wildcard*, enter the wildcard bits, in dotted decimal notation, to be applied to the source. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

SNMP Examples

This example shows how to enable all versions of SNMP. The configuration permits any SNMP manager to access all objects with read-only permissions using the community string *public*. This configuration does not cause the switch to send any traps.

```
Switch(config)# snmp-server community public
```

This example shows how to permit any SNMP manager to access all objects with read-only permission using the community string *public*. The switch also sends MAC notification traps to the hosts 192.180.1.111 and 192.180.1.33 using SNMPv1 and to the host 192.180.1.27 using SNMPv2C. The community string *public* is sent with the traps.

```
Switch(config)# snmp-server community public
Switch(config)# snmp-server enable traps mac-notification
Switch(config)# snmp-server host 192.180.1.27 version 2c public
Switch(config)# snmp-server host 192.180.1.111 version 1 public
Switch(config)# snmp-server host 192.180.1.33 public
```

This example shows how to allow read-only access for all objects to members of access list 4 that use the *comaccess* community string. No other SNMP managers have access to any objects. SNMP Authentication Failure traps are sent by SNMPv2C to the host *cisco.com* using the community string *public*.

```
Switch(config)# snmp-server community comaccess ro 4
Switch(config)# snmp-server enable traps snmp authentication
Switch(config)# snmp-server host cisco.com version 2c public
```

This example shows how to send Entity MIB traps to the host *cisco.com*. The community string is restricted. The first line enables the switch to send Entity MIB traps in addition to any traps previously enabled. The second line specifies the destination of these traps and overwrites any previous **snmp-server host** commands for the host *cisco.com*.

```
Switch(config)# snmp-server enable traps entity
Switch(config)# snmp-server host cisco.com restricted entity
```

This example shows how to enable the switch to send all traps to the host *myhost.cisco.com* using the community string *public*:

```
Switch(config)# snmp-server enable traps
Switch(config)# snmp-server host myhost.cisco.com public
```

This example shows how to associate a user with a remote host and to send **auth** (authNoPriv) authentication-level informs when the user enters global configuration mode:

```
Switch(config)# snmp-server engineID remote 192.180.1.27 00000063000100a1c0b4011b
Switch(config)# snmp-server group authgroup v3 auth
Switch(config)# snmp-server user authuser authgroup remote 192.180.1.27 v3 auth md5
mypassword
Switch(config)# snmp-server user authuser authgroup v3 auth md5 mypassword
Switch(config)# snmp-server host 192.180.1.27 informs version 3 auth authuser config
Switch(config)# snmp-server enable traps
Switch(config)# snmp-server inform retries 0
```

Displaying SNMP Status

To display SNMP input and output statistics, including the number of illegal community string entries, errors, and requested variables, use the **show snmp** privileged EXEC command. You also can use the other privileged EXEC commands in Table 27-6 to display SNMP information. For information about the fields in the displays, see the *Cisco IOS Configuration Fundamentals Command Reference, Release 12.2*.

Table 27-6 Commands for Displaying SNMP Information

Feature	Default Setting
show snmp	Displays SNMP statistics.
show snmp engineID [**local** \| **remote**]	Displays information on the local SNMP engine and all remote engines that have been configured on the device.
show snmp group	Displays information on each SNMP group on the network.
show snmp pending	Displays information on pending SNMP requests.
show snmp sessions	Displays information on the current SNMP sessions.
show snmp user	Displays information on each SNMP user name in the SNMP users table.



CHAPTER 28

Configuring Network Security with ACLs

This chapter describes how to configure network security on the Cisco ME 3400 Ethernet Access switch by using access control lists (ACLs), which are also referred to in commands and tables as access lists.

Note Information in this chapter about IP ACLs is specific to IP Version 4 (IPv4).

For complete syntax and usage information for the commands used in this chapter, see the command reference for this release, see the "Configuring IP Services" section in the "IP Addressing and Services" chapter of the *Cisco IOS IP Configuration Guide, Release 12.2*, and the *Cisco IOS IP Command Reference, Volume 1 of 3: Addressing and Services, Release 12.2*.

This chapter consists of these sections:

- Understanding ACLs, page 28-1
- Configuring IPv4 ACLs, page 28-6
- Creating Named MAC Extended ACLs, page 28-26
- Configuring VLAN Maps, page 28-29
- Using VLAN Maps with Router ACLs, page 28-36
- Displaying IPv4 ACL Configuration, page 28-39

Understanding ACLs

Packet filtering can help limit network traffic and restrict network use by certain users or devices. ACLs filter traffic as it passes through a router or switch and permit or deny packets crossing specified interfaces or VLANs. An ACL is a sequential collection of permit and deny conditions that apply to packets. When a packet is received on an interface, the switch compares the fields in the packet against any applied ACLs to verify that the packet has the required permissions to be forwarded, based on the criteria specified in the access lists. One by one, it tests packets against the conditions in an access list. The first match decides whether the switch accepts or rejects the packets. Because the switch stops testing after the first match, the order of conditions in the list is critical. If no conditions match, the switch rejects the packet. If there are no restrictions, the switch forwards the packet; otherwise, the switch drops the packet. The switch can use ACLs on all packets it forwards.

You configure access lists on a router or Layer 3 switch to provide basic security for your network. If you do not configure ACLs, all packets passing through the switch could be allowed onto all parts of the network. You can use ACLs to control which hosts can access different parts of a network or to decide

which types of traffic are forwarded or blocked at router interfaces. For example, you can allow e-mail traffic to be forwarded but not Telnet traffic. ACLs can be configured to block inbound traffic, outbound traffic, or both.

An ACL contains an ordered list of access control entries (ACEs). Each ACE specifies *permit* or *deny* and a set of conditions the packet must satisfy in order to match the ACE. The meaning of *permit* or *deny* depends on the context in which the ACL is used.

The switch supports IPv4 ACLs and Ethernet (MAC) ACLs:

- IP ACLs filter IPv4 traffic, including TCP, User Datagram Protocol (UDP), Internet Group Management Protocol (IGMP), and Internet Control Message Protocol (ICMP).
- Ethernet ACLs filter non-IP traffic.

This switch also supports quality of service (QoS) classification ACLs. For more information, see the "Understanding QoS" section on page 30-1.

These sections contain this conceptual information:

- Supported ACLs, page 28-2
- Handling Fragmented and Unfragmented Traffic, page 28-5

Supported ACLs

The switch supports three applications of ACLs to filter traffic:

- Port ACLs access-control traffic entering a Layer 2 interface. The switch does not support port ACLs in the outbound direction. You can apply only one IP access list and one MAC access list to a Layer 2 interface.
- Router ACLs access-control routed traffic between VLANs and are applied to Layer 3 interfaces in a specific direction (inbound or outbound). The switch must be running the metro IP access image to support router ACLs.
- VLAN ACLs or VLAN maps access-control all packets (forwarded and routed). You can use VLAN maps to filter traffic between devices in the same VLAN. VLAN maps are configured to provide access control based on Layer 3 addresses for IPv4. Unsupported protocols are access-controlled through MAC addresses using Ethernet ACEs. After a VLAN map is applied to a VLAN, all packets entering the VLAN are checked against the VLAN map. Packets can either enter the VLAN through a switch port or through a routed port after being routed.

You can use input port ACLs, router ACLs, and VLAN maps on the same switch. However, a port ACL takes precedence over a router ACL or VLAN map.

- When both an input port ACL and a VLAN map are applied, incoming packets received on ports with a port ACL applied are filtered by the port ACL. Other packets are filtered by the VLAN map
- When an input router ACL and input port ACL exist in an switch virtual interface (SVI), incoming packets received on ports to which a port ACL is applied are filtered by the port ACL. Incoming routed IPv4 packets received on other ports are filtered by the router ACL. Other packets are not filtered.
- When an output router ACL and input port ACL exist in an SVI, incoming packets received on the ports to which a port ACL is applied are filtered by the port ACL. Outgoing routed IPv4 packets are filtered by the router ACL. Other packets are not filtered.

- When a VLAN map, input router ACL, and input port ACL exist in an SVI, incoming packets received on the ports to which a port ACL is applied are only filtered by the port ACL. Incoming routed IPv4 packets received on other ports are filtered by both the VLAN map and the router ACL. Other packets are filtered only by the VLAN map.
- When a VLAN map, output router ACL, and input port ACL exist in an SVI, incoming packets received on the ports to which a port ACL is applied are only filtered by the port ACL. Outgoing routed IPv4 packets are filtered by both the VLAN map and the router ACL. Other packets are filtered only by the VLAN map.

If IEEE 802.1Q tunneling is configured on an interface, any IEEE 802.1Q encapsulated IPv4 packets received on the tunnel port can be filtered by MAC ACLs, but not by IP v4 ACLs. This is because the switch does not recognize the protocol inside the IEEE 802.1Q header. This restriction applies to router ACLs, port ACLs, and VLAN maps. For more information about IEEE 802.1Q tunneling, refer toChapter 13, “Configuring IEEE 802.1Q and Layer 2 Protocol Tunneling.”

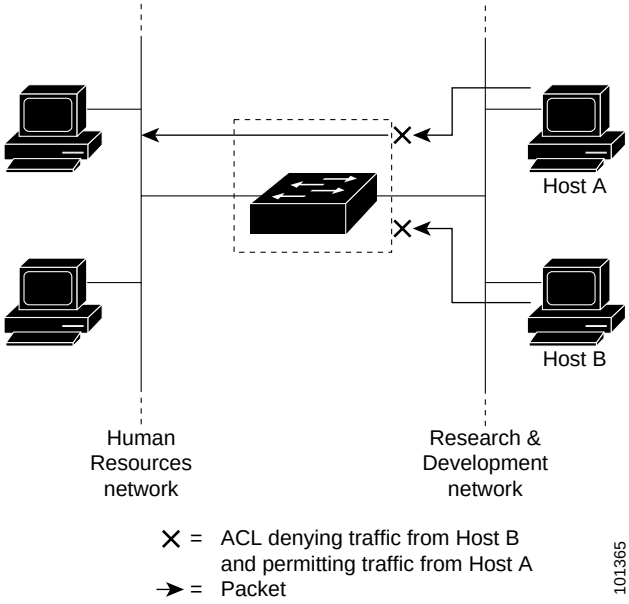
Port ACLs

Port ACLs are ACLs that are applied to Layer 2 interfaces on a switch. Port ACLs are supported only on physical interfaces and not on EtherChannel interfaces and can be applied only on interfaces in the inbound direction. These access lists are supported on Layer 2 interfaces:

- Standard IP access lists using source addresses
- Extended IP access lists using source and destination addresses and optional protocol type information
- MAC extended access lists using source and destination MAC addresses and optional protocol type information

The switch examines ACLs associated with all inbound features configured on a given interface and permits or denies packet forwarding based on how the packet matches the entries in the ACL. In this way, ACLs are used to control access to a network or to part of a network. Figure 28-1 is an example of using port ACLs to control access to a network when all workstations are in the same VLAN. ACLs applied at the Layer 2 input would allow Host A to access the Human Resources network, but prevent Host B from accessing the same network. Port ACLs can only be applied to Layer 2 interfaces in the inbound direction.

Figure 28-1 Using ACLs to Control Traffic to a Network



When you apply a port ACL to a trunk port, the ACL filters traffic on all VLANs present on the trunk port.

With port ACLs, you can filter IP traffic by using IP access lists and non-IP traffic by using MAC addresses. You can filter both IP and non-IP traffic on the same Layer 2 interface by applying both an IP access list and a MAC access list to the interface.

Note You cannot apply more than one IP access list and one MAC access list to a Layer 2 interface. If an IP access list or MAC access list is already configured on a Layer 2 interface and you apply a new IP access list or MAC access list to the interface, the new ACL replaces the previously configured one.

Router ACLs

If the switch is running the metro IP access image, you can apply router ACLs on switch virtual interfaces (SVIs), which are Layer 3 interfaces to VLANs; on physical Layer 3 interfaces; and on Layer 3 EtherChannel interfaces. You apply router ACLs on interfaces for specific directions (inbound or outbound). You can apply one router ACL in each direction on an interface.

One ACL can be used with multiple features for a given interface, and one feature can use multiple ACLs. When a single router ACL is used by multiple features, it is examined multiple times.

The switch supports these access lists for IPv4 traffic:

- Standard IP access lists use source addresses for matching operations.
- Extended IP access lists use source and destination addresses and optional protocol type information for matching operations.

As with port ACLs, the switch examines ACLs associated with features configured on a given interface. However, router ACLs are supported in both directions. As packets enter the switch on an interface, ACLs associated with all inbound features configured on that interface are examined. After packets are routed and before they are forwarded to the next hop, all ACLs associated with outbound features configured on the egress interface are examined.

ACLs permit or deny packet forwarding based on how the packet matches the entries in the ACL, and can be used to control access to a network or to part of a network. In Figure 28-1, ACLs applied at the router input allow Host A to access the Human Resources network, but prevent Host B from accessing the same network.

VLAN Maps

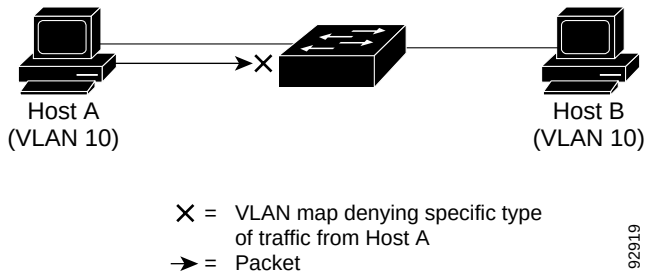
VLAN ACLs or VLAN maps can access-control *all* traffic. You can apply VLAN maps to all packets that are routed into or out of a VLAN or are forwarded within a VLAN in the switch. VLAN maps are used for security packet filtering and are not defined by direction (input or output).

You can configure VLAN maps to match Layer 3 addresses for IPv4 traffic.

All non-IP protocols are access-controlled through MAC addresses and Ethertype using MAC VLAN maps. (IP traffic *is not* access controlled by MAC VLAN maps.) You can enforce VLAN maps only on packets going through the switch; you cannot enforce VLAN maps on traffic between hosts on a hub or on another switch connected to this switch.

With VLAN maps, forwarding of packets is permitted or denied, based on the action specified in the map. Figure 28-2 shows how a VLAN map is applied to prevent a specific type of traffic from Host A in VLAN 10 from being forwarded. You can apply only one VLAN map to a VLAN.

Figure 28-2 Using VLAN Maps to Control Traffic



Handling Fragmented and Unfragmented Traffic

IPv4 packets can be fragmented as they cross the network. When this happens, only the fragment containing the beginning of the packet contains the Layer 4 information, such as TCP or UDP port numbers, ICMP type and code, and so on. All other fragments are missing this information.

Some ACEs do not check Layer 4 information and therefore can be applied to all packet fragments. ACEs that do test Layer 4 information cannot be applied in the standard manner to most of the fragments in a fragmented IPv4 packet. When the fragment contains no Layer 4 information and the ACE tests some Layer 4 information, the matching rules are modified:

- Permit ACEs that check the Layer 3 information in the fragment (including protocol type, such as TCP, UDP, and so on) are considered to match the fragment regardless of what the missing Layer 4 information might have been.
- Deny ACEs that check Layer 4 information never match a fragment unless the fragment contains Layer 4 information.

Consider access list 102, configured with these commands, applied to three fragmented packets:

```
Switch(config)# access-list 102 permit tcp any host 10.1.1.1 eq smtp
Switch(config)# access-list 102 deny tcp any host 10.1.1.2 eq telnet
Switch(config)# access-list 102 permit tcp any host 10.1.1.2
Switch(config)# access-list 102 deny tcp any any
```

Note In the first and second ACEs in the examples, the *eq* keyword after the destination address means to test for the TCP-destination-port well-known numbers equaling Simple Mail Transfer Protocol (SMTP) and Telnet, respectively.

- Packet A is a TCP packet from host 10.2.2.2., port 65000, going to host 10.1.1.1 on the SMTP port. If this packet is fragmented, the first fragment matches the first ACE (a permit) as if it were a complete packet because all Layer 4 information is present. The remaining fragments also match the first ACE, even though they do not contain the SMTP port information, because the first ACE only checks Layer 3 information when applied to fragments. The information in this example is that the packet is TCP and that the destination is 10.1.1.1.
- Packet B is from host 10.2.2.2, port 65001, going to host 10.1.1.2 on the Telnet port. If this packet is fragmented, the first fragment matches the second ACE (a deny) because all Layer 3 and Layer 4 information is present. The remaining fragments in the packet do not match the second ACE because they are missing Layer 4 information. Instead, they match the third ACE (a permit).

 Because the first fragment was denied, host 10.1.1.2 cannot reassemble a complete packet, so packet B is effectively denied. However, the later fragments that are permitted will consume bandwidth on the network and resources of host 10.1.1.2 as it tries to reassemble the packet.
- Fragmented packet C is from host 10.2.2.2, port 65001, going to host 10.1.1.3, port ftp. If this packet is fragmented, the first fragment matches the fourth ACE (a deny). All other fragments also match the fourth ACE because that ACE does not check any Layer 4 information and because Layer 3 information in all fragments shows that they are being sent to host 10.1.1.3, and the earlier permit ACEs were checking different hosts.

Configuring IPv4 ACLs

Configuring IP v4ACLs on the switch is the same as configuring IPv4 ACLs on other Cisco switches and routers. The process is briefly described here. For more detailed information on configuring ACLs, see the “Configuring IP Services” section in the “IP Addressing and Services” chapter of the *Cisco IOS IP Configuration Guide, Release 12.2*. For detailed information about the commands, see the *Cisco IOS IP Command Reference, Volume 1 of 3: Addressing and Services, Release 12.2*.

The switch does not support these Cisco IOS router ACL-related features:

- Non-IP protocol ACLs (see Table 28-1 on page 28-8) or bridge-group ACLs
- IP accounting
- Inbound and outbound rate limiting (except with QoS ACLs)
- Reflexive ACLs or dynamic ACLs
- ACL logging for port ACLs and VLAN maps

These are the steps to use IP ACLs on the switch:

Step 1 Create an ACL by specifying an access list number or name and the access conditions.

Step 2 Apply the ACL to interfaces or terminal lines. You can also apply standard and extended IP ACLs to VLAN maps.

These sections contain this configuration information:

- Creating Standard and Extended IPv4 ACLs, page 28-7
- Applying an IPv4 ACL to a Terminal Line, page 28-18
- Applying an IPv4 ACL to an Interface, page 28-19
- Hardware and Software Treatment of IP ACLs, page 28-21
- IPv4 ACL Configuration Examples, page 28-21

Creating Standard and Extended IPv4 ACLs

This section describes IP ACLs. An ACL is a sequential collection of permit and deny conditions. One by one, the switch tests packets against the conditions in an access list. The first match determines whether the switch accepts or rejects the packet. Because the switch stops testing after the first match, the order of the conditions is critical. If no conditions match, the switch denies the packet.

The software supports these types of ACLs or access lists for IPv4:

- Standard IP access lists use source addresses for matching operations.
- Extended IP access lists use source and destination addresses for matching operations and optional protocol-type information for finer granularity of control.

These sections describe access lists and how to create them:

- IPv4 Access List Numbers, page 28-8
- ACL Logging, page 28-8
- Creating a Numbered Standard ACL, page 28-9
- Creating a Numbered Extended ACL, page 28-10
- Resequencing ACEs in an ACL, page 28-14
- Creating Named Standard and Extended ACLs, page 28-14
- Using Time Ranges with ACLs, page 28-16
- Including Comments in ACLs, page 28-18

IPv4 Access List Numbers

The number you use to denote your IPv4 ACL shows the type of access list that you are creating. Table 28-1 lists the access-list number and corresponding access list type and shows whether or not they are supported in the switch. The switch supports IPv4 standard and extended access lists, numbers 1 to 199 and 1300 to 2699.

Table 28-1 Access List Numbers

Access List Number	Type	Supported
1–99	IP standard access list	Yes
100–199	IP extended access list	Yes
200–299	Protocol type-code access list	No
300–399	DECnet access list	No
400–499	XNS standard access list	No
500–599	XNS extended access list	No
600–699	AppleTalk access list	No
700–799	48-bit MAC address access list	No
800–899	IPX standard access list	No
900–999	IPX extended access list	No
1000–1099	IPX SAP access list	No
1100–1199	Extended 48-bit MAC address access list	No
1200–1299	IPX summary address access list	No
1300–1999	IP standard access list (expanded range)	Yes
2000–2699	IP extended access list (expanded range)	Yes

Note In addition to numbered standard and extended IPv4 ACLs, you can also create standard and extended named IPv4 ACLs by using the supported numbers. That is, the name of a standard IP ACL can be 1 to 99; the name of an extended IP ACL can be 100 to 199. The advantage of using named ACLs instead of numbered lists is that you can delete individual entries from a named list.

ACL Logging

The switch software can provide logging messages about packets permitted or denied by a standard IP access list. That is, any packet that matches the ACL causes an informational logging message about the packet to be sent to the console. The level of messages logged to the console is controlled by the logging console commands controlling the syslog messages.

Note Because routing is done in hardware and logging is done in software, if a large number of packets match a *permit* or *deny* ACE containing a **log** keyword, the software might not be able to match the hardware processing rate, and not all packets will be logged.

The first packet that triggers the ACL causes a logging message right away, and subsequent packets are collected over 5-minute intervals before they appear or logged. The logging message includes the access list number, whether the packet was permitted or denied, the source IP address of the packet, and the number of packets from that source permitted or denied in the prior 5-minute interval.

Creating a Numbered Standard ACL

Beginning in privileged EXEC mode, follow these steps to create a numbered standard ACL:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**access-list** *access-list-number* {**deny** \| **permit**} *source* [*source-wildcard*] [**log**]	Define a standard IPv4 access list by using a source address and wildcard. The *access-list-number* is a decimal number from 1 to 99 or 1300 to 1999. Enter **deny** or **permit** to specify whether to deny or permit access if conditions are matched. The *source* is the source address of the network or host from which the packet is being sent specified as: • The 32-bit quantity in dotted-decimal format. • The keyword **any** as an abbreviation for *source* and *source-wildcard* of 0.0.0.0 255.255.255.255. You do not need to enter a source-wildcard. • The keyword **host** as an abbreviation for source and source-wildcard of *source* 0.0.0.0. (Optional) The *source-wildcard* applies wildcard bits to the source. (Optional) Enter **log** to cause an informational logging message about the packet that matches the entry to be sent to the console.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show access-lists** [*number* \| *name*]	Show the access list configuration.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no access-list** *access-list-number* global configuration command to delete the entire ACL. You cannot delete individual ACEs from numbered access lists.

Note When creating an ACL, remember that, by default, the end of the ACL contains an implicit deny statement for all packets that it did not find a match for before reaching the end. With standard access lists, if you omit the mask from an associated IP host address ACL specification, 0.0.0.0 is assumed to be the mask.

This example shows how to create a standard ACL to deny access to IP host 171.69.198.102, permit access to any others, and display the results.

```
Switch (config)# access-list 2 deny host 171.69.198.102
Switch (config)# access-list 2 permit any
Switch(config)# end
Switch# show access-lists
Standard IP access list 2
    10 deny   171.69.198.102
    20 permit any
```

The switch always rewrites the order of standard access lists so that entries with **host** matches and entries with matches having a *don't care* mask of 0.0.0.0 are moved to the top of the list, above any entries with non-zero *don't care* masks. Therefore, in **show** command output and in the configuration file, the ACEs do not necessarily appear in the order in which they were entered.

After creating a numbered standard IPv4 ACL, you can apply it to terminal lines (see the "Applying an IPv4 ACL to a Terminal Line" section on page 28-18), to interfaces (see the "Applying an IPv4 ACL to an Interface" section on page 28-19), or to VLANs (see the "Configuring VLAN Maps" section on page 28-29).

Creating a Numbered Extended ACL

Although standard ACLs use only source addresses for matching, you can use extended ACL source and destination addresses for matching operations and optional protocol type information for finer granularity of control. When you are creating ACEs in numbered extended access lists, remember that after you create the ACL, any additions are placed at the end of the list. You cannot reorder the list or selectively add or remove ACEs from a numbered list.

Some protocols also have specific parameters and keywords that apply to that protocol.

These IP protocols are supported (protocol keywords are in parentheses in bold):

Authentication Header Protocol (**ahp**), Enhanced Interior Gateway Routing Protocol (**eigrp)**, Encapsulation Security Payload (**esp**), generic routing encapsulation (**gre**), Internet Control Message Protocol (**icmp)**, Internet Group Management Protocol (**igmp**), any Interior Protocol (**ip**), IP in IP tunneling (**ipinip**), KA9Q NOS-compatible IP over IP tunneling (**nos**), Open Shortest Path First routing (**ospf**), Payload Compression Protocol (**pcp**), Protocol Independent Multicast (**pim**), Transmission Control Protocol (**tcp**), or User Datagram Protocol (**udp**).

Note ICMP echo-reply cannot be filtered. All other ICMP codes or types can be filtered.

For more details on the specific keywords for each protocol, see these command references:

- *Cisco IOS IP Command Reference, Volume 1 of 3: Addressing and Services, Release 12.2*
- *Cisco IOS IP Command Reference, Volume 2 of 3: Routing Protocols, Release 12.2*
- *Cisco IOS IP Command Reference, Volume 3 of 3: Multicast, Release 12.2*

Note The switch does not support dynamic or reflexive access lists. It also does not support filtering based on the type of service (ToS) minimize-monetary-cost bit.

Supported parameters can be grouped into these categories: TCP, UDP, ICMP, IGMP, or other IP.

Beginning in privileged EXEC mode, follow these steps to create an extended ACL:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2a	**access-list** *access-list-number* {**deny** \| **permit**} *protocol source source-wildcard destination destination-wildcard* [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**log**] [**log-input**] [**time-range** *time-range-name*] [**dscp** *dscp*] **Note** If you enter a **dscp** value, you cannot enter **tos** or **precedence**. You can enter both a **tos** and a **precedence** value with no **dscp**.	Define an extended IPv4 access list and the access conditions. The *access-list-number* is a decimal number from 100 to 199 or 2000 to 2699. Enter **deny** or **permit** to specify whether to deny or permit the packet if conditions are matched. For *protocol*, enter the name or number of an IP protocol: **ahp**, **eigrp**, **esp**, **gre**, **icmp**, **igmp**, **igrp**, **ip**, **ipinip**, **nos**, **ospf**, **pcp**, **pim**, **tcp**, or **udp**, or an integer in the range 0 to 255 representing an IP protocol number. To match any Internet protocol (including ICMP, TCP, and UDP) use the keyword **ip**. **Note** This step includes options for most IP protocols. For additional specific parameters for TCP, UDP, ICMP, and IGMP, see steps 2b through 2e. The *source* is the number of the network or host from which the packet is sent. The *source-wildcard* applies wildcard bits to the source. The *destination* is the network or host number to which the packet is sent. The *destination-wildcard* applies wildcard bits to the destination. Source, source-wildcard, destination, and destination-wildcard can be specified as: • The 32-bit quantity in dotted-decimal format. • The keyword **any** for 0.0.0.0 255.255.255.255 (any host). • The keyword **host** for a single host 0.0.0.0. The other keywords are optional and have these meanings: • **precedence**—Enter to match packets with a precedence level specified as a number from 0 to 7 or by name: **routine** (**0**), ***priority*** (**1**), **immediate** (**2**), **flash** (**3**), **flash-override** (**4**), **critical** (**5**), **internet** (6), **network** (**7**). • **fragments**—Enter to check non-initial fragments. • **tos**—Enter to match by type of service level, specified by a number from 0 to 15 or a name: **normal** (**0**), **max-reliability** (**2**), **max-throughput** (**4**), **min-delay** (**8**). • **log**—Enter to create an informational logging message to be sent to the console about the packet that matches the entry or **log-input** to include the input interface in the log entry. • **time-range**—For an explanation of this keyword, see the "Using Time Ranges with ACLs" section on page 28-16. • **dscp**—Enter to match packets with the DSCP value specified by a number from 0 to 63, or use the question mark (?) to see a list of available values.

	Command	Purpose
or	**access-list** *access-list-number* {**deny** \| **permit**} *protocol* **any any** [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**log**] [**log-input**] [**time-range** *time-range-name*] [**dscp** *dscp*]	In access-list configuration mode, define an extended IP access list using an abbreviation for a source and source wildcard of 0.0.0.0 255.255.255.255 and an abbreviation for a destination and destination wildcard of 0.0.0.0 255.255.255.255. You can use the **any** keyword in place of source and destination address and wildcard.
or	**access-list** *access-list-number* {**deny** \| **permit**} *protocol* **host** *source* **host** *destination* [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**log**] [**log-input**] [**time-range** *time-range-name*] [**dscp** *dscp*]	Define an extended IP access list by using an abbreviation for a source and a source wildcard of *source* 0.0.0.0 and an abbreviation for a destination and destination wildcard of *destination* 0.0.0.0. You can use the **host** keyword in place of the source and destination wildcard or mask.
Step 2b	**access-list** *access-list-number* {**deny** \| **permit**} **tcp** *source source-wildcard* [*operator port*] *destination destination-wildcard* [*operator port*] [**established**] [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**log**] [**log-input**] [**time-range** *time-range-name*] [**dscp** *dscp*] [*flag*]	(Optional) Define an extended TCP access list and the access conditions. Enter **tcp** for Transmission Control Protocol. The parameters are the same as those described in Step 2a, with these exceptions: (Optional) Enter an *operator* and *port* to compare source (if positioned after *source source-wildcard*) or destination (if positioned after *destination destination-wildcard*) port. Possible operators include **eq** (equal), **gt** (greater than), **lt** (less than), **neq** (not equal), and **range** (inclusive range). Operators require a port number (**range** requires two port numbers separated by a space). Enter the *port* number as a decimal number (from 0 to 65535) or the name of a TCP port. To see TCP port names, use the ? or see the "Configuring IP Services" section in the "IP Addressing and Services" chapter of the *Cisco IOS IP Configuration Guide, Release 12.2*. Use only TCP port numbers or names when filtering TCP. The other optional keywords have these meanings: • **established**—Enter to match an established connection. This has the same function as matching on the **ack** or **rst** flag. • *flag*—Enter one of these flags to match by the specified TCP header bits: **ack** (acknowledge), **fin** (finish), **psh** (push), **rst** (reset), **syn** (synchronize), or **urg** (urgent).
Step 2c	**access-list** *access-list-number* {**deny** \| **permit**} **udp** *source source-wildcard* [*operator port*] *destination destination-wildcard* [*operator port*] [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**log**] [**log-input**] [**time-range** *time-range-name*] [**dscp** *dscp*]	(Optional) Define an extended UDP access list and the access conditions. Enter **udp** for the User Datagram Protocol. The UDP parameters are the same as those described for TCP except that the [*operator* [*port*]] port number or name must be a UDP port number or name, and the **flag** and **established** parameters are not valid for UDP.

	Command	Purpose
Step 2d	**access-list** *access-list-number* {**deny** \| **permit**} **icmp** *source source-wildcard destination destination-wildcard* [*icmp-type* \| [[*icmp-type icmp-code*] \| [*icmp-message*]] [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**log**] [**log-input**] [**time-range** *time-range-name*] [**dscp** *dscp*]	(Optional) Define an extended ICMP access list and the access conditions. Enter **icmp** for Internet Control Message Protocol. The ICMP parameters are the same as those described for most IP protocols in Step 2a, with the addition of the ICMP message type and code parameters. These optional keywords have these meanings: • *icmp-type*—Enter to filter by ICMP message type, a number from 0 to 255. • *icmp-code*—Enter to filter ICMP packets that are filtered by the ICMP message code type, a number from 0 to 255. • *icmp-message*—Enter to filter ICMP packets by the ICMP message type name or the ICMP message type and code name. To see a list of ICMP message type names and code names, use the ?, or see the "Configuring IP Services" section of the *Cisco IOS IP Configuration Guide, Release 12.2*.
Step 2e	**access-list** *access-list-number* {**deny** \| **permit**} **igmp** *source source-wildcard destination destination-wildcard* [*igmp-type*] [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**log**] [**log-input**] [**time-range** *time-range-name*] [**dscp** *dscp*]	(Optional) Define an extended IGMP access list and the access conditions. Enter **igmp** for Internet Group Management Protocol. The IGMP parameters are the same as those described for most IP protocols in Step 2a, with this optional parameter. *igmp-type*—To match IGMP message type, enter a number from 0 to 15, or enter the message name (**host-query**, **host-report**, **pim,** or **trace**). **Note** Although visible in the command-line help, **dvmrp** is not supported.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show access-lists** [*number* \| *name*]	Verify the access list configuration.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no access-list** *access-list-number* global configuration command to delete the entire access list. You cannot delete individual ACEs from numbered access lists.

This example shows how to create and display an extended access list to deny Telnet access from any host in network 171.69.198.0 to any host in network 172.20.52.0 and to permit any others. (The **eq** keyword after the destination address means to test for the TCP destination port number equaling Telnet.)

```
Switch(config)# access-list 102 deny tcp 171.69.198.0 0.0.0.255 172.20.52.0 0.0.0.255 eq
telnet
Switch(config)# access-list 102 permit tcp any any
Switch(config)# end
Switch# show access-lists
Extended IP access list 102
    10 deny tcp 171.69.198.0 0.0.0.255 172.20.52.0 0.0.0.255 eq telnet
    20 permit tcp any any
```

After an ACL is created, any additions (possibly entered from the terminal) are placed at the end of the list. You cannot selectively add or remove access list entries from a numbered access list.

Note When you are creating an ACL, remember that, by default, the end of the access list contains an implicit deny statement for all packets if it did not find a match before reaching the end.

After creating a numbered extended ACL, you can apply it to terminal lines (see the "Applying an IPv4 ACL to a Terminal Line" section on page 28-18), to interfaces (see the "Applying an IPv4 ACL to an Interface" section on page 28-19), or to VLANs (see the "Configuring VLAN Maps" section on page 28-29).

Resequencing ACEs in an ACL

Sequence numbers for the entries in an access list are automatically generated when you create a new ACL.You can use the **ip access-list resequence** global configuration command to edit the sequence numbers in an ACL and change the order in which ACEs are applied. For example, if you add a new ACE to an ACL, it is placed at the bottom of the list. By changing the sequence number, you can move the ACE to a different position in the ACL.

For more information about the **ip access-list resequence** command, see this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122s/122snwft/release/122s14/fsaclseq.htm

Creating Named Standard and Extended ACLs

You can identify IPv4 ACLs with an alphanumeric string (a name) rather than a number. You can use named ACLs to configure more IPv4 access lists in a router than if you were to use numbered access lists. If you identify your access list with a name rather than a number, the mode and command syntax are slightly different. However, not all commands that use IP access lists accept a named access list.

Note The name you give to a standard or extended ACL can also be a number in the supported range of access list numbers. That is, the name of a standard IP ACL can be 1 to 99; the name of an extended IP ACL can be 100 to 199. The advantage of using named ACLs instead of numbered lists is that you can delete individual entries from a named list.

Consider these guidelines and limitations before configuring named ACLs:

- Not all commands that accept a numbered ACL accept a named ACL. ACLs for packet filters and route filters on interfaces can use a name. VLAN maps also accept a name.
- A standard ACL and an extended ACL cannot have the same name.
- Numbered ACLs are also available, as described in the "Creating Standard and Extended IPv4 ACLs" section on page 28-7.
- You can use standard and extended ACLs (named or numbered) in VLAN maps.

Beginning in privileged EXEC mode, follow these steps to create a standard ACL using names:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip access-list standard** *name*	Define a standard IPv4 access list using a name, and enter access-list configuration mode. **Note** The name can be a number from 1 to 99.

	Command	Purpose
Step 3	**deny** {*source* [*source-wildcard*] \| **host** *source* \| **any**} [**log**] or **permit** {*source* [*source-wildcard*] \| **host** *source* \| **any**} [**log**]	In access-list configuration mode, specify one or more conditions denied or permitted to decide if the packet is forwarded or dropped. • **host** *source*—A source and source wildcard of *source* 0.0.0.0. • **any**—A source and source wildcard of 0.0.0.0 255.255.255.255.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show access-lists** [*number* \| *name*]	Show the access list configuration.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove a named standard ACL, use the **no ip access-list standard** *name* global configuration command.

Beginning in privileged EXEC mode, follow these steps to create an extended ACL using names:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip access-list extended** *name*	Define an extended IPv4 access list using a name and enter access-list configuration mode. **Note** The name can be a number from 100 to 199.
Step 3	{**deny** \| **permit**} *protocol* {*source* [*source-wildcard*] \| **host** *source* \| **any**} {*destination* [*destination-wildcard*] \| **host** *destination* \| **any**} [**precedence** *precedence*] [**tos** *tos*] [**established**] [**log**] [**time-range** *time-range-name*]	In access-list configuration mode, specify the conditions allowed or denied. Use the **log** keyword to get access list logging messages, including violations. See the "Creating a Numbered Extended ACL" section on page 28-10 for definitions of protocols and other keywords. • **host** *source*—A source and source wildcard of *source* 0.0.0.0. • ***host*** *destination*—A destination and destination wildcard of *destination* 0.0.0.0. • **any**—A source and source wildcard or destination and destination wildcard of 0.0.0.0 255.255.255.255.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show access-lists** [*number* \| *name*]	Show the access list configuration.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove a named extended ACL, use the **no ip access-list extended** *name* global configuration command.

When you are creating standard extended ACLs, remember that, by default, the end of the ACL contains an implicit deny statement for everything if it did not find a match before reaching the end. For standard ACLs, if you omit the mask from an associated IP host address access list specification, 0.0.0.0 is assumed to be the mask.

After you create an ACL, any additions are placed at the end of the list. You cannot selectively add ACL entries to a specific ACL. However, you can use **no permit** and **no deny** access-list configuration mode commands to remove entries from a named ACL.

This example shows how you can delete individual ACEs from the named access list *border-list*:

```
Switch(config)# ip access-list extended border-list
Switch(config-ext-nacl)# no permit ip host 10.1.1.3 any
```

Being able to selectively remove lines from a named ACL is one reason you might use named ACLs instead of numbered ACLs.

After creating a named ACL, you can apply it to interfaces (see the "Applying an IPv4 ACL to an Interface" section on page 28-19) or to VLANs (see the "Configuring VLAN Maps" section on page 28-29).

Using Time Ranges with ACLs

You can selectively apply extended ACLs based on the time of day and week by using the **time-range** global configuration command. First, define a time-range name and set the times and the dates or the days of the week in the time range. Then enter the time-range name when applying an ACL to set restrictions to the access list. You can use the time range to define when the permit or deny statements in the ACL are in effect, for example, during a specified time period or on specified days of the week. The **time-range** keyword and argument are referenced in the named and numbered extended ACL task tables in the previous sections, the "Creating Standard and Extended IPv4 ACLs" section on page 28-7, and the "Creating Named Standard and Extended ACLs" section on page 28-14.

These are some of the many possible benefits of using time ranges:

- You have more control over permitting or denying a user access to resources, such as an application (identified by an IP address/mask pair and a port number).
- You can control logging messages. ACL entries can be set to log traffic only at certain times of the day. Therefore, you can simply deny access without needing to analyze many logs generated during peak hours.

Time-based access lists trigger CPU activity because the new configuration of the access list must be merged with other features and the combined configuration loaded into the TCAM. For this reason, you should be careful not to have several access lists configured to take affect in close succession (within a small number of minutes of each other.)

Note The time range relies on the switch system clock; therefore, you need a reliable clock source. We recommend that you use Network Time Protocol (NTP) to synchronize the switch clock. For more information, see the "Managing the System Time and Date" section on page 5-1.

Beginning in privileged EXEC mode, follow these steps to configure a time-range parameter for an ACL:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**time-range** *time-range-name*	Assign a meaningful name (for example, *workhours*) to the time range to be created, and enter time-range configuration mode. The name cannot contain a space or quotation mark and must begin with a letter.

	Command	Purpose
Step 3	**absolute** [**start** *time date*] [**end** *time date*] or **periodic** *day-of-the-week hh:mm to* [*day-of-the-week*] *hh:mm* or **periodic** {**weekdays** \| **weekend** \| **daily**} *hh:mm to hh:mm*	Specify when the function it will be applied to is operational. • You can use only one **absolute** statement in the time range. If you configure more than one absolute statement, only the one configured last is executed. • You can enter multiple **periodic** statements. For example, you could configure different hours for weekdays and weekends. See the example configurations.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show time-range**	Verify the time-range configuration.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Repeat the steps if you have multiple items that you want in effect at different times.

To remove a configured time-range limitation, use the **no time-range** *time-range-name* global configuration command.

This example shows how to configure time ranges for *workhours* and to configure January 1, 2006 as a company holiday and to verify your configuration.

```
Switch(config)# time-range workhours
Switch(config-time-range)# periodic weekdays 8:00 to 12:00
Switch(config-time-range)# periodic weekdays 13:00 to 17:00
Switch(config-time-range)# exit
Switch(config)# time-range new_year_day_2006
Switch(config-time-range)# absolute start 00:00 1 Jan 2006 end 23:59 1 Jan 2006
Switch(config-time-range)# end
Switch# show time-range
time-range entry: new_year_day_2003 (inactive)
   absolute start 00:00 01 January 2006 end 23:59 01 January 2006
time-range entry: workhours (inactive)
   periodic weekdays 8:00 to 12:00
   periodic weekdays 13:00 to 17:00
```

To apply a time-range, enter the time-range name in an extended ACL that can implement time ranges. This example shows how to create and verify extended access list 188 that denies TCP traffic from any source to any destination during the defined holiday times and permits all TCP traffic during work hours.

```
Switch(config)# access-list 188 deny tcp any any time-range new_year_day_2006
Switch(config)# access-list 188 permit tcp any any time-range workhours
Switch(config)# end
Switch# show access-lists
Extended IP access list 188
   10 deny tcp any any time-range new_year_day_2006 (inactive)
   20 permit tcp any any time-range workhours (inactive)
```

This example uses named ACLs to permit and deny the same traffic.

```
Switch(config)# ip access-list extended deny_access
Switch(config-ext-nacl)# deny tcp any any time-range new_year_day_2006
Switch(config-ext-nacl)# exit
Switch(config)# ip access-list extended may_access
Switch(config-ext-nacl)# permit tcp any any time-range workhours
Switch(config-ext-nacl)# end
Switch# show ip access-lists
Extended IP access list deny_access
    10 deny tcp any any time-range new_year_day_2006 (inactive)
Extended IP access list may_access
    10 permit tcp any any time-range workhours (inactive)
```

Including Comments in ACLs

You can use the **remark** keyword to include comments (remarks) about entries in any IP standard or extended ACL. The remarks make the ACL easier for you to understand and scan. Each remark line is limited to 100 characters.

The remark can go before or after a permit or deny statement. You should be consistent about where you put the remark so that it is clear which remark describes which permit or deny statement. For example, it would be confusing to have some remarks before the associated permit or deny statements and some remarks after the associated statements.

To include a comment for IP numbered standard or extended ACLs, use the **access-list** *access-list number* **remark** *remark* global configuration command. To remove the remark, use the **no** form of this command.

In this example, the workstation that belongs to Jones is allowed access, and the workstation that belongs to Smith is not allowed access:

```
Switch(config)# access-list 1 remark Permit only Jones workstation through
Switch(config)# access-list 1 permit 171.69.2.88
Switch(config)# access-list 1 remark Do not allow Smith through
Switch(config)# access-list 1 deny 171.69.3.13
```

For an entry in a named IP ACL, use the **remark** access-list configuration command. To remove the remark, use the **no** form of this command.

In this example, the Jones subnet is not allowed to use outbound Telnet:

```
Switch(config)# ip access-list extended telnetting
Switch(config-ext-nacl)# remark Do not allow Jones subnet to telnet out
Switch(config-ext-nacl)# deny tcp host 171.69.2.88 any eq telnet
```

Applying an IPv4 ACL to a Terminal Line

You can use numbered ACLs to control access to one or more terminal lines. You cannot apply named ACLs to lines. You must set identical restrictions on all the virtual terminal lines because a user can attempt to connect to any of them.

For procedures for applying ACLs to interfaces, see the "Applying an IPv4 ACL to an Interface" section on page 28-19. For applying ACLs to VLANs, see the "Configuring VLAN Maps" section on page 28-29.

Beginning in privileged EXEC mode, follow these steps to restrict incoming and outgoing connections between a virtual terminal line and the addresses in an ACL:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**line** [**console** \| **vty**] *line-number*	Identify a specific line to configure, and enter in-line configuration mode. • **console**—Specify the console terminal line. The console port is DCE. • **vty**—Specify a virtual terminal for remote console access. The *line-number* is the first line number in a contiguous group that you want to configure when the line type is specified. The range is from 0 to 16.
Step 3	**access-class** *access-list-number* {**in** \| **out**}	Restrict incoming and outgoing connections between a particular virtual terminal line (into a device) and the addresses in an access list.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Display the access list configuration.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove an ACL from a terminal line, use the **no access-class** *access-list-number* {**in** | **out**} line configuration command.

Applying an IPv4 ACL to an Interface

This section describes how to apply IPv4 ACLs to network interfaces. You can apply an ACL to *either* outbound or inbound Layer 3 interfaces. You can apply ACLs only to inbound Layer 2 interfaces. Note these guidelines:

- When controlling access to an interface, you can use a named or numbered ACL.
- If you apply an ACL to a Layer 2 interface that is a member of a VLAN, the Layer 2 (port) ACL takes precedence over an input Layer 3 ACL applied to the VLAN interface or a VLAN map applied to the VLAN. Incoming packets received on the Layer 2 port are always filtered by the port ACL.
- If you apply an ACL to a Layer 3 interface and routing is not enabled on the switch, the ACL only filters packets that are intended for the CPU, such as SNMP, Telnet, or web traffic. You do not have to enable routing to apply ACLs to Layer 2 interfaces.
- When private VLANs are configured, you can apply router ACLs only on the primary-VLAN SVIs. The ACL is applied to both primary and secondary VLAN Layer 3 traffic.

Note By default, the router sends Internet Control Message Protocol (ICMP) unreachable messages when a packet is denied by an access group. These access-group denied packets are not dropped in hardware but are bridged to the switch CPU so that it can generate the ICMP-unreachable message.

Beginning in privileged EXEC mode, follow these steps to control access to an interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Identify a specific interface for configuration, and enter interface configuration mode. The interface can be a Layer 2 interface (port ACL), or a Layer 3 interface (router ACL).
Step 3	**no shutdown**	Enable the port, if necessary. By default, user network interfaces (UNIs) are disabled, and network node interfaces (NNIs) are enabled.
Step 4	**ip access-group** {*access-list-number* \| *name*} {**in** \| **out**}	Control access to the specified interface. The **out** keyword is not supported for Layer 2 interfaces (port ACLs).
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config**	Display the access list configuration.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the specified access group, use the **no ip access-group** {*access-list-number* | *name*} {**in** | **out**} interface configuration command.

This example shows how to apply access list 2 to a port to filter packets entering the port:

```
Switch(config)# interface gigabitethernet0/1
Router(config-if)# ip access-group 2 in
```

Note When you apply the **ip access-group** interface configuration command to a Layer 3 interface (an SVI, a Layer 3 EtherChannel, or a routed port), the interface must have been configured with an IP address. Layer 3 access groups filter packets that are routed or are received by Layer 3 processes on the CPU.

For inbound ACLs, after receiving a packet, the switch checks the packet against the ACL. If the ACL permits the packet, the switch continues to process the packet. If the ACL rejects the packet, the switch discards the packet.

For outbound ACLs, after receiving and routing a packet to a controlled interface, the switch checks the packet against the ACL. If the ACL permits the packet, the switch sends the packet. If the ACL rejects the packet, the switch discards the packet.

By default, the input interface sends ICMP Unreachable messages whenever a packet is discarded, regardless of whether the packet was discarded because of an ACL on the input interface or because of an ACL on the output interface. ICMP Unreachables are normally limited to no more than one every one-half second per input interface, but this can be changed by using the **ip icmp rate-limit unreachable** global configuration command.

When you apply an undefined ACL to an interface, the switch acts as if the ACL has not been applied to the interface and permits all packets. Remember this behavior if you use undefined ACLs for network security.

Hardware and Software Treatment of IP ACLs

ACL processing is primarily accomplished in hardware, but requires forwarding of some traffic flows to the CPU for software processing. If the hardware reaches its capacity to store ACL configurations, packets are sent to the CPU for forwarding. The forwarding rate for software-forwarded traffic is substantially less than for hardware-forwarded traffic.

Note If an ACL configuration cannot be implemented in hardware due to an out-of-resource condition on a switch, then only the traffic in that VLAN arriving on that switch is affected (forwarded in software). Software forwarding of packets might adversely impact the performance of the switch, depending on the number of CPU cycles that this consumes.

For router ACLs, other factors can cause packets to be sent to the CPU:

- Using the **log** keyword
- Generating ICMP unreachable messages

When traffic flows are both logged and forwarded, forwarding is done by hardware, but logging must be done by software. Because of the difference in packet handling capacity between hardware and software, if the sum of all flows being logged (both permitted flows and denied flows) is of great enough bandwidth, not all of the packets that are forwarded can be logged.

If router ACL configuration cannot be applied in hardware, packets arriving in a VLAN that must be routed are routed in software. If ACLs cause large numbers of packets to be sent to the CPU, the switch performance can be negatively affected.

When you enter the **show ip access-lists** privileged EXEC command, the match count displayed does not account for packets that are access controlled in hardware. Use the **show access-lists hardware counters** privileged EXEC command to obtain some basic hardware ACL statistics for switched and routed packets.

Router ACLs function as follows:

- The hardware controls permit and deny actions of standard and extended ACLs (input and output) for security access control.
- If **log** has not been specified, the flows that match a *deny* statement in a security ACL are dropped by the hardware if *ip unreachables* is disabled. The flows matching a *permit* statement are switched in hardware.
- Adding the **log** keyword to an ACE in a router ACL causes a copy of the packet to be sent to the CPU for logging only. If the ACE is a *permit* statement, the packet is still switched and routed in hardware.

IPv4 ACL Configuration Examples

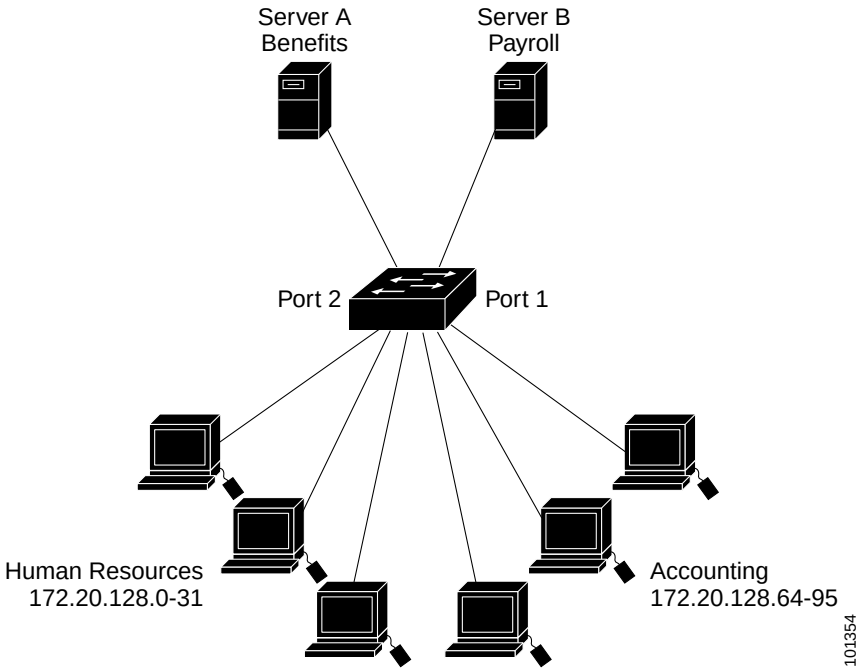
This section provides examples of configuring and applying IPv4 ACLs. For detailed information about compiling ACLs, see the *Cisco IOS Security Configuration Guide, Release 12.2* and to the Configuring IP Services" section in the "IP Addressing and Services" chapter of the *Cisco IOS IP Configuration Guide, Release 12.2.*

Figure 28-3 shows a small networked office environment with routed Port 2 connected to Server A, containing benefits and other information that all employees can access, and routed Port 1 connected to Server B, containing confidential payroll data. All users can access Server A, but Server B has restricted access.

Use router ACLs to do this in one of two ways:

- Create a standard ACL, and filter traffic coming to the server from Port 1.
- Create an extended ACL, and filter traffic coming from the server into Port 1.

Figure 28-3 Using Router ACLs to Control Traffic



This example uses a standard ACL to filter traffic coming into Server B from a port, permitting traffic only from Accounting's source addresses 172.20.128.64 to 172.20.128.95. The ACL is applied to traffic coming out of routed Port 1 from the specified source address.

```
Switch(config)# access-list 6 permit 172.20.128.64 0.0.0.31
Switch(config)# end
Switch# show access-lists
Standard IP access list 6
    10 permit 172.20.128.64, wildcard bits 0.0.0.31
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ip access-group 6 out
```

This example uses an extended ACL to filter traffic coming from Server B into a port, permitting traffic from any source address (in this case Server B) to only the Accounting destination addresses 172.20.128.64 to 172.20.128.95. The ACL is applied to traffic going into routed Port 1, permitting it to go only to the specified destination addresses. Note that with extended ACLs, you must enter the protocol (IP) before the source and destination information.

```
Switch(config)# access-list 106 permit ip any 172.20.128.64 0.0.0.31
Switch(config)# end
Switch# show access-lists
Extended IP access list 106
    10 permit ip any 172.20.128.64 0.0.0.31
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ip access-group 106 in
```

Numbered ACLs

In this example, network 36.0.0.0 is a Class A network whose second octet specifies a subnet; that is, its subnet mask is 255.255.0.0. The third and fourth octets of a network 36.0.0.0 address specify a particular host. Using access list 2, the switch accepts one address on subnet 48 and reject all others on that subnet. The last line of the list shows that the switch accepts addresses on all other network 36.0.0.0 subnets. The ACL is applied to packets entering a port.

```
Switch(config)# access-list 2 permit 36.48.0.3
Switch(config)# access-list 2 deny 36.48.0.0 0.0.255.255
Switch(config)# access-list 2 permit 36.0.0.0 0.255.255.255
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ip access-group 2 in
```

Extended ACLs

In this example, the first line permits any incoming TCP connections with destination ports greater than 1023. The second line permits incoming TCP connections to the Simple Mail Transfer Protocol (SMTP) port of host 128.88.1.2. The third line permits incoming ICMP messages for error feedback.

```
Switch(config)# access-list 102 permit tcp any 128.88.0.0 0.0.255.255 gt 1023
Switch(config)# access-list 102 permit tcp any host 128.88.1.2 eq 25
Switch(config)# access-list 102 permit icmp any any
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ip access-group 102 in
```

For another example of using an extended ACL, suppose that you have a network connected to the Internet, and you want any host on the network to be able to form TCP connections to any host on the Internet. However, you do not want IP hosts to be able to form TCP connections to hosts on your network, except to the mail (SMTP) port of a dedicated mail host.

SMTP uses TCP port 25 on one end of the connection and a random port number on the other end. The same port numbers are used throughout the life of the connection. Mail packets coming in from the Internet have a destination port of 25. Outbound packets have the port numbers reversed. Because the secure system of the network always accepts mail connections on port 25, the incoming and outgoing services are separately controlled. The ACL must be configured as an input ACL on the outbound interface and an output ACL on the inbound interface.

In this example, the network is a Class B network with the address 128.88.0.0, and the mail host address is 128.88.1.2. The **established** keyword is used only for the TCP to show an established connection. A match occurs if the TCP datagram has the ACK or RST bits set, which show that the packet belongs to an existing connection. Gigabit Ethernet interface 1 is the interface that connects the router to the Internet.

```
Switch(config)# access-list 102 permit tcp any 128.88.0.0 0.0.255.255 established
Switch(config)# access-list 102 permit tcp any host 128.88.1.2 eq 25
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ip access-group 102 in
```

Named ACLs

This example creates a standard ACL named *internet_filter* and an extended ACL named *marketing_group*. The *internet_filter* ACL allows all traffic from the source address 1.2.3.4.

```
Switch(config)# ip access-list standard Internet_filter
Switch(config-ext-nacl)# permit 1.2.3.4
Switch(config-ext-nacl)# exit
```

The *marketing_group* ACL allows any TCP Telnet traffic to the destination address and wildcard 171.69.0.0 0.0.255.255 and denies any other TCP traffic. It permits ICMP traffic, denies UDP traffic from any source to the destination address range 171.69.0.0 through 179.69.255.255 with a destination port less than 1024, denies any other IP traffic, and provides a log of the result.

```
Switch(config)# ip access-list extended marketing_group
Switch(config-ext-nacl)# permit tcp any 171.69.0.0 0.0.255.255 eq telnet
Switch(config-ext-nacl)# deny tcp any any
Switch(config-ext-nacl)# permit icmp any any
Switch(config-ext-nacl)# deny udp any 171.69.0.0 0.0.255.255 lt 1024
Switch(config-ext-nacl)# deny ip any any log
Switch(config-ext-nacl)# exit
```

The *Internet_filter* ACL is applied to outgoing traffic and the *marketing_group* ACL is applied to incoming traffic on a Layer 3 port.

```
Switch(config)# interface gigabitethernet0/2
Switch(config-if)# no switchport
Switch(config-if)# ip address 2.0.5.1 255.255.255.0
Switch(config-if)# ip access-group Internet_filter out
Switch(config-if)# ip access-group marketing_group in
```

Time Range Applied to an IP ACL

This example denies HTTP traffic on IP on Monday through Friday between the hours of 8:00 a.m. and 6:00 p.m (18:00). The example allows UDP traffic only on Saturday and Sunday from noon to 8:00 p.m. (20:00).

```
Switch(config)# time-range no-http
Switch(config)# periodic weekdays 8:00 to 18:00
!
Switch(config)# time-range udp-yes
Switch(config)# periodic weekend 12:00 to 20:00
!
Switch(config)# ip access-list extended strict
Switch(config-ext-nacl)# deny tcp any any eq www time-range no-http
Switch(config-ext-nacl)# permit udp any any time-range udp-yes
!
Switch(config-ext-nacl)# exit
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ip access-group strict in
```

Commented IP ACL Entries

In this example of a numbered ACL, the workstation that belongs to Jones is allowed access, and the workstation that belongs to Smith is not allowed access:

```
Switch(config)# access-list 1 remark Permit only Jones workstation through
Switch(config)# access-list 1 permit 171.69.2.88
Switch(config)# access-list 1 remark Do not allow Smith workstation through
Switch(config)# access-list 1 deny 171.69.3.13
```

In this example of a numbered ACL, the Winter and Smith workstations are not allowed to browse the web:

```
Switch(config)# access-list 100 remark Do not allow Winter to browse the web
Switch(config)# access-list 100 deny host 171.69.3.85 any eq www
Switch(config)# access-list 100 remark Do not allow Smith to browse the web
Switch(config)# access-list 100 deny host 171.69.3.13 any eq www
```

In this example of a named ACL, the Jones subnet is not allowed access:

```
Switch(config)# ip access-list standard prevention
Switch(config-std-nacl)# remark Do not allow Jones subnet through
Switch(config-std-nacl)# deny 171.69.0.0 0.0.255.255
```

In this example of a named ACL, the Jones subnet is not allowed to use outbound Telnet:

```
Switch(config)# ip access-list extended telnetting
Switch(config-ext-nacl)# remark Do not allow Jones subnet to telnet out
Switch(config-ext-nacl)# deny tcp 171.69.0.0 0.0.255.255 any eq telnet
```

ACL Logging

Two variations of logging are supported on router ACLs. The **log** keyword sends an informational logging message to the console about the packet that matches the entry; the **log-input** keyword includes the input interface in the log entry.

In this example, standard named access list *stan1* denies traffic from 10.1.1.0 0.0.0.255, allows traffic from all other sources, and includes the **log** keyword.

```
Switch(config)# ip access-list standard stan1
Switch(config-std-nacl)# deny 10.1.1.0 0.0.0.255 log
Switch(config-std-nacl)# permit any log
Switch(config-std-nacl)# exit
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ip access-group stan1 in
Switch(config-if)# end
Switch# show logging
Syslog logging: enabled (0 messages dropped, 0 flushes, 0 overruns)
    Console logging: level debugging, 37 messages logged
    Monitor logging: level debugging, 0 messages logged
    Buffer logging: level debugging, 37 messages logged
    File logging: disabled
    Trap logging: level debugging, 39 message lines logged

Log Buffer (4096 bytes):

00:00:48: NTP: authentication delay calculation problems

<output truncated>

00:09:34:%SEC-6-IPACCESSLOGS:list stan1 permitted 0.0.0.0 1 packet
00:09:59:%SEC-6-IPACCESSLOGS:list stan1 denied 10.1.1.15 1 packet
00:10:11:%SEC-6-IPACCESSLOGS:list stan1 permitted 0.0.0.0 1 packet
```

This example is a named extended access list *ext1* that permits ICMP packets from any source to 10.1.1.0 0.0.0.255 and denies all UDP packets.

```
Switch(config)# ip access-list extended ext1
Switch(config-ext-nacl)# permit icmp any 10.1.1.0 0.0.0.255 log
Switch(config-ext-nacl)# deny udp any any log
Switch(config-std-nacl)# exit
Switch(config)# interface gigabitethernet0/2
Switch(config-if)# ip access-group ext1 in
```

This is a an example of a log for an extended ACL:

```
01:24:23:%SEC-6-IPACCESSLOGDP:list ext1 permitted icmp 10.1.1.15 -> 10.1.1.61 (0/0), 1
packet
01:25:14:%SEC-6-IPACCESSLOGDP:list ext1 permitted icmp 10.1.1.15 -> 10.1.1.61 (0/0), 7
packets
01:26:12:%SEC-6-IPACCESSLOGP:list ext1 denied udp 0.0.0.0(0) -> 255.255.255.255(0), 1
packet
01:31:33:%SEC-6-IPACCESSLOGP:list ext1 denied udp 0.0.0.0(0) -> 255.255.255.255(0), 8
packets
```

Note that all logging entries for IP ACLs start with `%SEC-6-IPACCESSLOG` with minor variations in format depending on the kind of ACL and the access entry that has been matched.

This is an example of an output message when the **log-input** keyword is entered:

```
00:04:21:%SEC-6-IPACCESSLOGDP:list inputlog permitted icmp 10.1.1.10 (Vlan1
0001.42ef.a400) -> 10.1.1.61 (0/0), 1 packet
```

A log message for the same sort of packet using the **log** keyword does not include the input interface information:

```
00:05:47:%SEC-6-IPACCESSLOGDP:list inputlog permitted icmp 10.1.1.10 -> 10.1.1.61 (0/0), 1
packet
```

Creating Named MAC Extended ACLs

You can filter non-IPv4 traffic on a VLAN or on a Layer 2 interface by using MAC addresses and named MAC extended ACLs. The procedure is similar to that of configuring other extended named ACLs.

Note You cannot apply named MAC extended ACLs to Layer 3 interfaces.

For more information about the supported non-IP protocols in the **mac access-list extended** command, see the command reference for this release.

Note Though visible in the command-line help strings, **appletalk** is not supported as a matching condition for the **deny** and **permit** MAC access-list configuration mode commands.

Beginning in privileged EXEC mode, follow these steps to create a named MAC extended ACL:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mac access-list extended** *name*	Define an extended MAC access list using a name.

	Command	Purpose
Step 3	{**deny** \| **permit**} {**any** \| **host** *source MAC address* \| *source MAC address mask*} {**any** \| **host** *destination MAC address* \| *destination MAC address mask*} [*type mask* \| **lsap** *lsap mask* \| **aarp** \| **amber** \| **dec-spanning** \| **decnet-iv** \| **diagnostic** \| **dsm** \| **etype-6000** \| **etype-8042** \| **lat** \| **lavc-sca** \| **mop-console** \| **mop-dump** \| **msdos** \| **mumps** \| **netbios** \| **vines-echo** \|**vines-ip** \| **xns-idp** \| *0-65535*] [**cos** *cos*]	In extended MAC access-list configuration mode, specify to **permit** or **deny any** source MAC address, a source MAC address with a mask, or a specific **host** source MAC address and **any** destination MAC address, destination MAC address with a mask, or a specific destination MAC address. (Optional) You can also enter these options: • *type mask*—An arbitrary EtherType number of a packet with Ethernet II or SNAP encapsulation in decimal, hexadecimal, or octal with optional mask of *don't care* bits applied to the EtherType before testing for a match. • **lsap** *lsap mask*—An LSAP number of a packet with IEEE 802.2 encapsulation in decimal, hexadecimal, or octal with optional mask of *don't care* bits. • **aarp** \| **amber** \| **dec-spanning** \| **decnet-iv** \| **diagnostic** \| **dsm** \| **etype-6000** \| **etype-8042** \| **lat** \| **lavc-sca** \| **mop-console** \| **mop-dump** \| **msdos** \| **mumps** \| **netbios** \| **vines-echo** \|**vines-ip** \| **xns-idp**—A non-IP protocol. • **cos** *cos*—An IEEE 802.1Q cost of service number from 0 to 7 used to set priority.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show access-lists** [*number* \| *name*]	Show the access list configuration.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no mac access-list extended** *name* global configuration command to delete the entire ACL. You can also delete individual ACEs from named MAC extended ACLs.

This example shows how to create and display an access list named *mac1*, denying only EtherType DECnet Phase IV traffic, but permitting all other types of traffic.

```
Switch(config)# mac access-list extended mac1
Switch(config-ext-macl)# deny any any decnet-iv
Switch(config-ext-macl)# permit any any
Switch(config-ext-macl)# end
Switch # show access-lists
Extended MAC access list mac1
    10 deny   any any decnet-iv
    20 permit any any
```

Applying a MAC ACL to a Layer 2 Interface

After you create a MAC ACL, you can apply it to a Layer 2 interface to filter non-IP traffic coming in that interface. When you apply the MAC ACL, consider these guidelines:

- If you apply an ACL to a Layer 2 interface that is a member of a VLAN, the Layer 2 (port) ACL takes precedence over an input Layer 3 ACL applied to the VLAN interface or a VLAN map applied to the VLAN. Incoming packets received on the Layer 2 port are always filtered by the port ACL.
- You can apply no more than one IP access list and one MAC access list to the same Layer 2 interface. The IP access list filters only IP packets, and the MAC access list filters non-IP packets.
- A Layer 2 interface can have only one MAC access list. If you apply a MAC access list to a Layer 2 interface that has a MAC ACL configured, the new ACL replaces the previously configured one.

Beginning in privileged EXEC mode, follow these steps to apply a MAC access list to control access to a Layer 2 interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Identify a specific interface, and enter interface configuration mode. The interface must be a physical Layer 2 interface (port ACL).
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled, and NNIs are enabled.
Step 4	**mac access-group** {*name*} {**in**}	Control access to the specified interface by using the MAC access list. **Note** Port ACLs are supported only in the inbound direction.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show mac access-group** [**interface** *interface-i*d]	Display the MAC access list applied to the interface or all Layer 2 interfaces.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the specified access group, use the **no mac access-group** {*name*} interface configuration command.

This example shows how to apply MAC access list *mac1* to a port to filter packets entering the port:

```
Switch(config)# interface gigabitethernet0/2
Router(config-if)# mac access-group mac1 in
```

Note The **mac access-group** interface configuration command is only valid when applied to a physical Layer 2 interface.You cannot use the command on EtherChannel port channels.

After receiving a packet, the switch checks it against the inbound ACL. If the ACL permits it, the switch continues to process the packet. If the ACL rejects the packet, the switch discards it. When you apply an undefined ACL to an interface, the switch acts as if the ACL has not been applied and permits all packets. Remember this behavior if you use undefined ACLs for network security.

Configuring VLAN Maps

This section describes how to configure VLAN maps, which is the only way to control filtering within a VLAN. VLAN maps have no direction. To filter traffic in a specific direction by using a VLAN map, you need to include an ACL with specific source or destination addresses. If there is a match clause for that type of packet (IP or MAC) in the VLAN map, the default action is to drop the packet if the packet does not match any of the entries within the map. If there is no match clause for that type of packet, the default is to forward the packet.

Note For complete syntax and usage information for the commands used in this section, see the command reference for this release.

To create a VLAN map and apply it to one or more VLANs, perform these steps:

Step 1 Create the standard or extended IPv4 ACLs or named MAC extended ACLs that you want to apply to the VLAN. See the "Creating Standard and Extended IPv4 ACLs" section on page 28-7 and the "Creating a VLAN Map" section on page 28-31.

Step 2 Enter the **vlan access-map** global configuration command to create a VLAN ACL map entry.

Step 3 In access-map configuration mode, optionally enter an **action**—**forward** (the default) or **drop**—and enter the **match** command to specify an IP packet or a non-IP packet (with only a known MAC address) and to match the packet against one or more ACLs (standard or extended).

Note If the VLAN map has a match clause for the type of packet (IP or MAC) and the packet does not match the type, the default is to drop the packet. If there is no match clause in the VLAN map for that type of packet, and no action specified, the packet is forwarded.

Step 4 Use the **vlan filter** global configuration command to apply a VLAN map to one or more VLANs.

These sections contain this configuration information:

- VLAN Map Configuration Guidelines, page 28-30
- Creating a VLAN Map, page 28-31
- Applying a VLAN Map to a VLAN, page 28-33
- Using VLAN Maps in Your Network, page 28-34

VLAN Map Configuration Guidelines

Follow these guidelines when configuring VLAN maps:

- If there is no ACL configured to deny traffic on an interface and *no* VLAN map is configured, all traffic is permitted.
- Each VLAN map consists of a series of entries. The order of entries in an VLAN map is important. A packet that comes into the switch is tested against the first entry in the VLAN map. If it matches, the action specified for that part of the VLAN map is taken. If there is no match, the packet is tested against the next entry in the map.
- If the VLAN map has at least one match clause for the type of packet (IP or MAC) and the packet does not match any of these match clauses, the default is to drop the packet. If there is no match clause for that type of packet in the VLAN map, the default is to forward the packet.
- The system might take longer to boot if you have configured a very large number of ACLs.
- Logging is not supported for VLAN maps.
- If VLAN map configuration cannot be applied in hardware, all packets in that VLAN must be routed by software.
- When a switch has an IP access list or MAC access list applied to a Layer 2 interface, and you apply a VLAN map to a VLAN that the port belongs to, the port ACL takes precedence over the VLAN map.
- You can configure VLAN maps on primary and secondary VLANs. However, we recommend that you configure the same VLAN maps on private-VLAN primary and secondary VLANs.
- When a frame is Layer-2 forwarded within a private VLAN, the same VLAN map is applied at the ingress side and at the egress side. When a frame is routed from inside a private VLAN to an external port, the private-VLAN map is applied at the ingress side.
 - For frames going upstream from a host port to a promiscuous port, the VLAN map configured on the secondary VLAN is applied.
 - For frames going downstream from a promiscuous port to a host port, the VLAN map configured on the primary VLAN is applied.

 To filter out specific IP traffic for a private VLAN, you should apply the VLAN map to both the primary and secondary VLANs. For more information about private VLANs, see Chapter 12, “Configuring Private VLANs.”
- See the “Using VLAN Maps in Your Network” section on page 28-34 for configuration examples.
- For information about using both router ACLs and VLAN maps, see the “VLAN Maps and Router ACL Configuration Guidelines” section on page 28-36.

Creating a VLAN Map

Each VLAN map consists of an ordered series of entries. Beginning in privileged EXEC mode, follow these steps to create, add to, or delete a VLAN map entry:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**vlan access-map** *name* [*number*]	Create a VLAN map, and give it a name and (optionally) a number. The number is the sequence number of the entry within the map. When you create VLAN maps with the same name, numbers are assigned sequentially in increments of 10. When modifying or deleting maps, you can enter the number of the map entry that you want to modify or delete. Entering this command changes to access-map configuration mode.
Step 3	**action** {**drop** \| **forward**}	(Optional) Set the action for the map entry. The default is to forward.
Step 4	**match** {**ip** \| **mac**} **address** {*name* \| *number*} [*name* \| *number*]	Match the packet (using either the IP or MAC address) against one or more standard or extended access lists. Note that packets are only matched against access lists of the correct protocol type. IP packets are matched against standard or extended IP access lists. Non-IP packets are only matched against named MAC extended access lists.
Step 5	**end**	Return to global configuration mode.
Step 6	**show running-config**	Display the access list configuration.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no vlan access-map** *name* global configuration command to delete a map.

Use the **no vlan access-map** *name number* global configuration command to delete a single sequence entry from within the map.

Use the **no action** access-map configuration command to enforce the default action, which is to forward.

VLAN maps do not use the specific permit or deny keywords. To deny a packet by using VLAN maps, create an ACL that would match the packet, and set the action to drop. A permit in the ACL counts as a match. A deny in the ACL means no match.

Examples of ACLs and VLAN Maps

These examples show how to create ACLs and VLAN maps that for specific purposes.

Example 1

This example shows how to create an ACL and a VLAN map to deny a packet. In the first map, any packets that match the *ip1* ACL (TCP packets) would be dropped. You first create the *ip1*ACL to permit any TCP packet and no other packets. Because there is a match clause for IP packets in the VLAN map, the default action is to drop any IP packet that does not match any of the match clauses.

```
Switch(config)# ip access-list extended ip1
Switch(config-ext-nacl)# permit tcp any any
Switch(config-ext-nacl)# exit
Switch(config)# vlan access-map map_1 10
Switch(config-access-map)# match ip address ip1
Switch(config-access-map)# action drop
```

This example shows how to create a VLAN map to permit a packet. ACL *ip2* permits UDP packets and any packets that match the *ip2* ACL are forwarded. In this map, any IP packets that did not match any of the previous ACLs (that is, packets that are not TCP packets or UDP packets) would get dropped.

```
Switch(config)# ip access-list extended ip2
Switch(config-ext-nacl)# permit udp any any
Switch(config-ext-nacl)# exit
Switch(config)# vlan access-map map_1 20
Switch(config-access-map)# match ip address ip2
Switch(config-access-map)# action forward
```

Example 2

In this example, the VLAN map has a default action of drop for IP packets and a default action of forward for MAC packets. Used with standard ACL 101 and extended named access lists **igmp-match** and **tcp-match**, the map will have the following results:

- Forward all UDP packets
- Drop all IGMP packets
- Forward all TCP packets
- Drop all other IP packets
- Forward all non-IP packets

```
Switch(config)# access-list 101 permit udp any any
Switch(config)# ip access-list extended igmp-match
Switch(config-ext-nacl)# permit igmp any any
Switch(config)# ip access-list extended tcp-match
Switch(config-ext-nacl)# permit tcp any any
Switch(config-ext-nacl)# exit
Switch(config)# vlan access-map drop-ip-default 10
Switch(config-access-map)# match ip address 101
Switch(config-access-map)# action forward
Switch(config-access-map)# exit
Switch(config)# vlan access-map drop-ip-default 20
Switch(config-access-map)# match ip address igmp-match
Switch(config-access-map)# action drop
Switch(config-access-map)# exit
Switch(config)# vlan access-map drop-ip-default 30
Switch(config-access-map)# match ip address tcp-match
Switch(config-access-map)# action forward
```

Example 3

In this example, the VLAN map has a default action of drop for MAC packets and a default action of forward for IP packets. Used with MAC extended access lists **good-hosts** and **good-protocols**, the map will have the following results:

- Forward MAC packets from hosts 0000.0c00.0111 and 0000.0c00.0211
- Forward MAC packets with decnet-iv or vines-ip protocols
- Drop all other non-IP packets
- Forward all IP packets

```
Switch(config)# mac access-list extended good-hosts
Switch(config-ext-macl)# permit host 000.0c00.0111 any
Switch(config-ext-macl)# permit host 000.0c00.0211 any
Switch(config-ext-nacl)# exit
```

```
Switch(config)# mac access-list extended good-protocols
Switch(config-ext-macl)# permit any any decnet-ip
Switch(config-ext-macl)# permit any any vines-ip
Switch(config-ext-nacl)# exit
Switch(config)# vlan access-map drop-mac-default 10
Switch(config-access-map)# match mac address good-hosts
Switch(config-access-map)# action forward
Switch(config-access-map)# exit
Switch(config)# vlan access-map drop-mac-default 20
Switch(config-access-map)# match mac address good-protocols
Switch(config-access-map)# action forward
```

Example 4

In this example, the VLAN map has a default action of drop for all packets (IP and non-IP). Used with access lists **tcp-match** and **good-hosts** from Examples 2 and 3, the map will have the following results:

- Forward all TCP packets
- Forward MAC packets from hosts 0000.0c00.0111 and 0000.0c00.0211
- Drop all other IP packets
- Drop all other MAC packets

```
Switch(config)# vlan access-map drop-all-default 10
Switch(config-access-map)# match ip address tcp-match
Switch(config-access-map)# action forward
Switch(config-access-map)# exit
Switch(config)# vlan access-map drop-all-default 20
Switch(config-access-map)# match mac address good-hosts
Switch(config-access-map)# action forward
```

Applying a VLAN Map to a VLAN

Beginning in privileged EXEC mode, follow these steps to apply a VLAN map to one or more VLANs:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**vlan filter** *mapname* **vlan-list** *list*	Apply the VLAN map to one or more VLAN IDs. The list can be a single VLAN ID (22), a consecutive list (10-22), or a string of VLAN IDs (12, 22, 30). Spaces around the comma and hyphen are optional.
Step 3	**show running-config**	Display the access list configuration.
Step 4	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the VLAN map, use the **no vlan filter** *mapname* **vlan-list** *list* global configuration command.

This example shows how to apply VLAN map 1 to VLANs 20 through 22:

```
Switch(config)# vlan filter map 1 vlan-list 20-22
```

Using VLAN Maps in Your Network

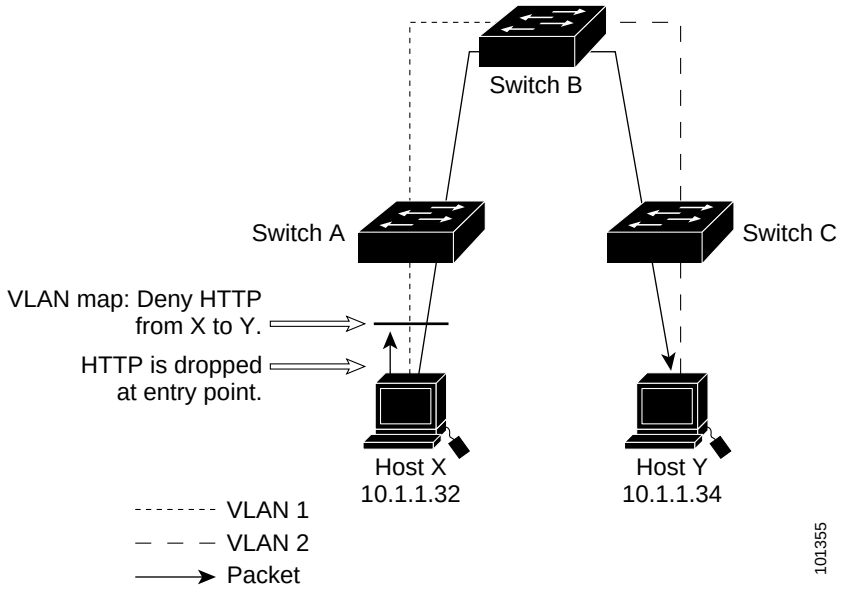
These sections describes some typical uses for VLAN maps:

- Wiring Closet Configuration, page 28-34
- Denying Access to a Server on Another VLAN, page 28-35

Wiring Closet Configuration

In a wiring closet configuration, routing might not be enabled on the switch. In this configuration, the switch can still support a VLAN map and a QoS classification ACL. In Figure 28-4, assume that Host X and Host Y are in different VLANs and are connected to wiring closet switches A and C. Traffic from Host X to Host Y is eventually being routed by Switch B, a Layer 3 switch with routing enabled. Traffic from Host X to Host Y can be access-controlled at the traffic entry point, Switch A.

Figure 28-4 Wiring Closet Configuration



If you do not want HTTP traffic switched from Host X to Host Y, you can configure a VLAN map on Switch A to drop all HTTP traffic from Host X (IP address 10.1.1.32) to Host Y (IP address 10.1.1.34) at Switch A and not forward it to Switch B.

First, define the IP access list *http* that permits (matches) any TCP traffic on the HTTP port.

```
Switch(config)# ip access-list extended http
Switch(config-ext-nacl)# permit tcp host 10.1.1.32 host 10.1.1.34 eq www
Switch(config-ext-nacl)# exit
```

Next, create VLAN access map *map2* so that traffic that matches the *http* access list is dropped and all other IP traffic is forwarded.

```
Switch(config)# vlan access-map map2 10
Switch(config-access-map)# match ip address http
Switch(config-access-map)# action drop
Switch(config-access-map)# exit
```

```
Switch(config)# ip access-list extended match_all
Switch(config-ext-nacl)# permit ip any any
Switch(config-ext-nacl)# exit
Switch(config)# vlan access-map map2 20
Switch(config-access-map)# match ip address match_all
Switch(config-access-map)# action forward
```

Then, apply VLAN access map *map2* to VLAN 1.

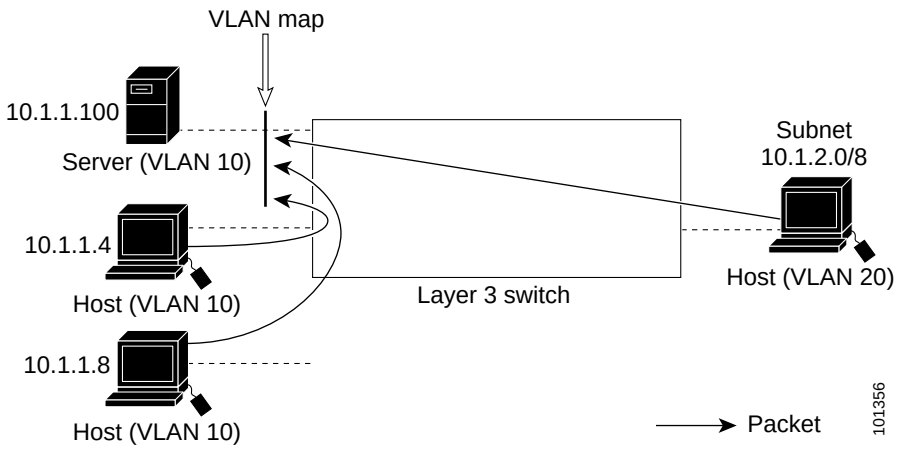
```
Switch(config)# vlan filter map2 vlan 1
```

Denying Access to a Server on Another VLAN

You can restrict access to a server on another VLAN. For example, server 10.1.1.100 in VLAN 10 needs to have access denied to these hosts (see Figure 28-5):

- Hosts in subnet 10.1.2.0/8 in VLAN 20 should not have access.
- Hosts 10.1.1.4 and 10.1.1.8 in VLAN 10 should not have access.

Figure 28-5 Deny Access to a Server on Another VLAN



This example shows how to deny access to a server on another VLAN by creating the VLAN map SERVER 1 that denies access to hosts in subnet 10.1.2.0.8, host 10.1.1.4, and host 10.1.1.8 and permits other IP traffic. The final step is to apply the map SERVER1 to VLAN 10.

Step 1 Define the IP ACL that will match the correct packets.

```
Switch(config)# ip access-list extended SERVER1_ACL
Switch(config-ext-nacl))# permit ip 10.1.2.0 0.0.0.255 host 10.1.1.100
Switch(config-ext-nacl))# permit ip host 10.1.1.4 host 10.1.1.100
Switch(config-ext-nacl))# permit ip host 10.1.1.8 host 10.1.1.100
Switch(config-ext-nacl))# exit
```

Step 2 Define a VLAN map using this ACL that will drop IP packets that match SERVER1_ACL and forward IP packets that do not match the ACL.

```
Switch(config)# vlan access-map SERVER1_MAP
Switch(config-access-map)# match ip address SERVER1_ACL
Switch(config-access-map)# action drop
Switch(config)# vlan access-map SERVER1_MAP 20
Switch(config-access-map)# action forward
Switch(config-access-map)# exit
```

Step 3 Apply the VLAN map to VLAN 10.

```
Switch(config)# vlan filter SERVER1_MAP vlan-list 10.
```

Using VLAN Maps with Router ACLs

To access control routed traffic, you can use VLAN maps only or a combination of router ACLs and VLAN maps. You can define router ACLs on both input and output routed VLAN interfaces. If a packet flow matches a VLAN-map deny clause in the ACL, regardless of the router ACL configuration, the packet flow is denied.

Note When you use router ACLs with VLAN maps, packets that require logging on the router ACLs are not logged if they are denied by a VLAN map.

If the VLAN map has a match clause for the type of packet (IP or MAC) and the packet does not match the type, the default is to drop the packet. If there is no match clause in the VLAN map, and no action specified, the packet is forwarded if it does not match any VLAN map entry.

These sections contain information about using VLAN maps with router ACLs:

- VLAN Maps and Router ACL Configuration Guidelines, page 28-36
- Examples of Router ACLs and VLAN Maps Applied to VLANs, page 28-37

VLAN Maps and Router ACL Configuration Guidelines

These guidelines are for configurations where you need to have an router ACL *and* a VLAN map on the same VLAN. These guidelines do not apply to configurations where you are mapping router ACLs and VLAN maps on different VLANs.

The switch hardware provides one lookup for security ACLs for each direction (input and output); therefore, you must merge a router ACL and a VLAN map when they are configured on the same VLAN. Merging the router ACL with the VLAN map might significantly increase the number of ACEs.

If you must configure a router ACL and a VLAN map on the same VLAN, use these guidelines for both router ACL and VLAN map configuration:

- You can configure only one VLAN map and one router ACL in each direction (input/output) on a VLAN interface.

- Whenever possible, try to write the ACL with all entries having a single action except for the final, default action of the other type. That is, write the ACL using one of these two forms:

 permit...
 permit...
 permit...
 deny ip any any

 or

 deny...
 deny...
 deny...
 permit ip any any

- To define multiple actions in an ACL (permit, deny), group each action type together to reduce the number of entries.
- Avoid including Layer 4 information in an ACL; adding this information complicates the merging process. The best merge results are obtained if the ACLs are filtered based on IP addresses (source and destination) and not on the full flow (source IP address, destination IP address, protocol, and protocol ports). It is also helpful to use *don't care* bits in the IP address, whenever possible.

 If you need to specify the full-flow mode and the ACL contains both IP ACEs and TCP/UDP/ICMP ACEs with Layer 4 information, put the Layer 4 ACEs at the end of the list. This gives priority to the filtering of traffic based on IP addresses.

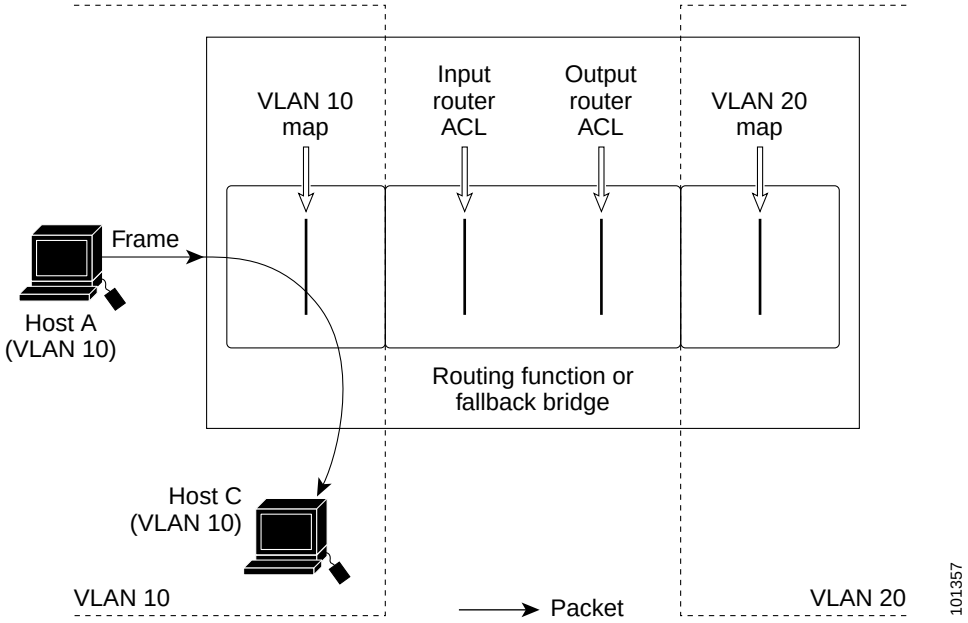
Examples of Router ACLs and VLAN Maps Applied to VLANs

This section gives examples of applying router ACLs and VLAN maps to a VLAN for switched, routed, and multicast packets. Although the following illustrations show packets being forwarded to their destination, each time the packet's path crosses a line indicating a VLAN map or an ACL, it is also possible that the packet might be dropped, rather than forwarded.

ACLs and Switched Packets

Figure 28-6 shows how an ACL is applied on packets that are switched within a VLAN. Packets switched within the VLAN without being routed or forwarded are only subject to the VLAN map of the input VLAN.

Figure 28-6 Applying ACLs on Switched Packets

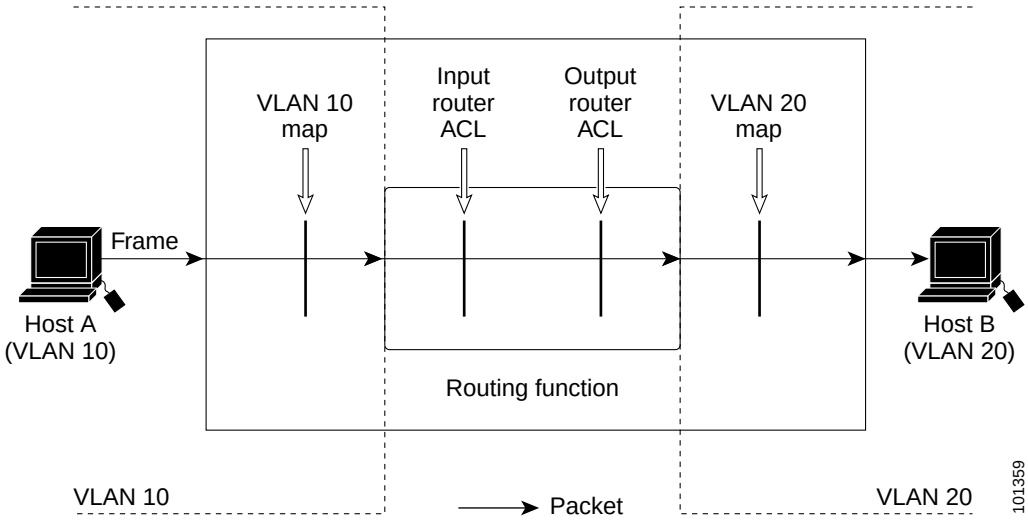


ACLs and Routed Packets

Figure 28-7 shows how ACLs are applied on routed packets. For routed packets, the ACLs are applied in this order:

1. VLAN map for input VLAN
2. Input router ACL
3. Output router ACL
4. VLAN map for output VLAN

Figure 28-7 Applying ACLs on Routed Packets

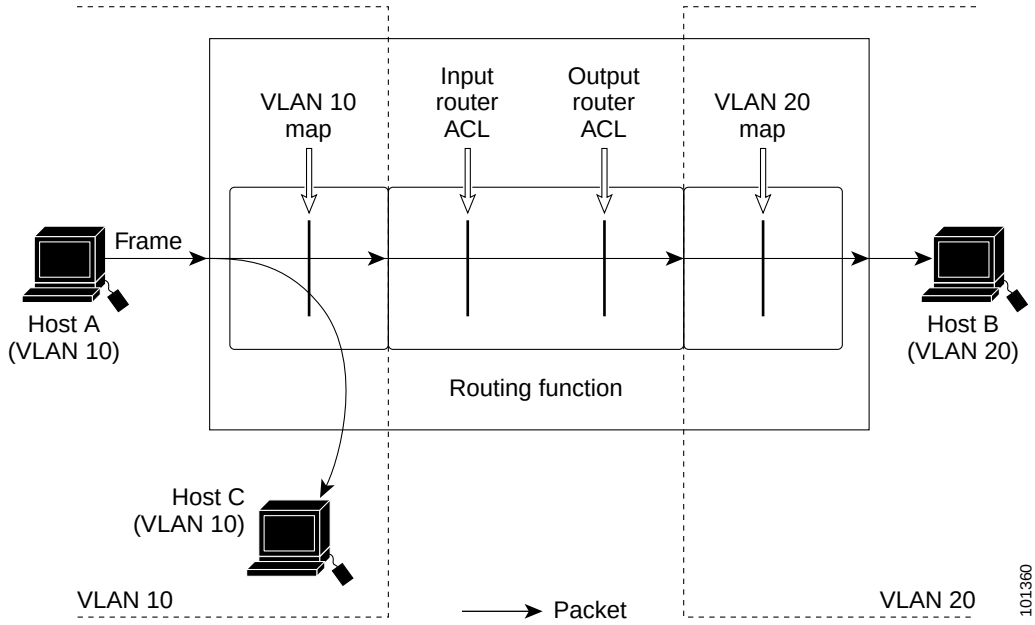


ACLs and Multicast Packets

Figure 28-8 shows how ACLs are applied on packets that are replicated for IP multicasting. A multicast packet being routed has two different kinds of filters applied: one for destinations that are other ports in the input VLAN and another for each of the destinations that are in other VLANs to which the packet has been routed. The packet might be routed to more than one output VLAN, in which case a different router output ACL and VLAN map would apply for each destination VLAN.

The final result is that the packet might be permitted in some of the output VLANs and not in others. A copy of the packet is forwarded to those destinations where it is permitted. However, if the input VLAN map (VLAN 10 map in Figure 28-8) drops the packet, no destination receives a copy of the packet.

Figure 28-8 Applying ACLs on Multicast Packets



Displaying IPv4 ACL Configuration

You can display the ACLs that are configured on the switch, and you can display the ACLs that have been applied to interfaces and VLANs.

When you use the **ip access-group** interface configuration command to apply ACLs to a Layer 2 or 3 interface, you can display the access groups on the interface. You can also display the MAC ACLs applied to a Layer 2 interface. You can use the privileged EXEC commands as described in Table 28-2 to display this information.

Table 28-2 Commands for Displaying Access Lists and Access Groups

Command	Purpose
show access-lists [*number* \| *name*]	Displays the contents of one or all current IP and MAC address access lists or a specific access list (numbered or named).
show ip access-lists [*number* \| *name*]	Displays the contents of all current IP access lists or a specific IP access list (numbered or named).

Table 28-2 Commands for Displaying Access Lists and Access Groups (continued)

Command	Purpose
show ip interface *interface-id*	Displays detailed configuration and status of an interface. If IP is enabled on the interface and ACLs have been applied by using the **ip access-group** interface configuration command, the access groups are included in the display.
show running-config [**interface** *interface-id*]	Displays the contents of the configuration file for the switch or the specified interface, including all configured MAC and IP access lists and which access groups are applied to an interface.
show mac access-group [**interface** *interface-i*d]	Displays MAC access lists applied to all Layer 2 interfaces or the specified Layer 2 interface.

You can also display information about VLAN access maps or VLAN filters. Use the privileged EXEC commands in Table 28-3 to display VLAN map information.

Table 28-3 Commands for Displaying VLAN Map Information

Command	Purpose
show vlan access-map [*mapname*]	Shows information about all VLAN access-maps or the specified access map.
show vlan filter [**access-map** *name* \| **vlan** *vlan-id*]	Shows information about all VLAN filters or about a specified VLAN or VLAN access map.



Configuring Control-Plane Security

This chapter describes the control-plane security feature in the Cisco ME 3400 Ethernet Access switch. In any network, Layer 2 and Layer 3 switches exchange control packets with other switches in the network. The Cisco ME switch, which acts as a transition between the customer network and the service-provider network, uses control-plane security to ensure that the topology information between the two networks is isolated. This mechanism protects against a possible denial-of-service attack from another customer network.

This chapter includes these sections;

- Understanding Control-Plane Security, page 29-1
- Configuring Control-Plane Security, page 29-4
- Monitoring Control-Plane Security, page 29-5

Understanding Control-Plane Security

The Cisco ME switch can have no more than four ports configured as network node interfaces (NNIs) that connect to the service-provider network. The switch communicates with the rest of the network through these ports, exchanging protocol control packets as well as regular traffic. The remainder of the ports on the Cisco ME switch are user network interfaces (UNIs) that are used as customer-facing ports. Each port is connected to a single customer, and exchanging network protocol control packets between the switch and the customer is not usually required. To protect against accidental or intentional CPU overload, the Cisco ME switch provides control-plane security automatically by dropping or rate-limiting a predefined set of Layer 2 control packets and some Layer 3 control packets for UNIs.

Control-plane security is supported on a port for Layer 2 control packets and non-IP packets with router MAC addresses regardless of whether the port is in routing or nonrouting mode. (A port is in routing mode when global IP routing is enabled and it is configured with the **no switchport** interface configuration command or associated with a VLAN that has a switch virtual interface [SVI] created and active.) These packets are either dropped or rate-limited, depending upon the Layer 2 protocol configuration. For Layer 3 control packets, on a port in routing mode (whether or not a Layer 3 service policy is attached), control-plane security supports rate-limiting only Internet Group Management Protocol (IGMP) control packets. For Layer 3 packets, on a port in non-routing mode (whether or not a Layer 2 service policy is attached), only IP packets with router MAC addresses are dropped.

These types of control packets are dropped or rate-limited:

- Layer 2 protocol control packets:
 - Control packets that are always dropped on UNIs, such as Dynamic Trunking Protocol (DTP) packets and some bridge protocol data units (BPDUs).
 - Control packets that are dropped by default but can be enabled or tunneled, such as Cisco Discovery Protocol (CDP), Spanning-Tree Protocol (STP), VLAN Trunking Protocol (VTP), UniDirectional Link Detection (UDLD) protocol, Link Aggregation Control Protocol (LACP), and Port Aggregation Protocol (PAgP) packets. When enabled, these protocol packets are rate-limited and tunneled through the switch.
 - Control or management packets that are required by the switch, such as keepalive packets. These control packets are processed by the CPU but rate-limited to normal and safe limits to prevent CPU overload.
- Non-IP packets with router MAC addresses
- IP packets with router MAC addresses
- IGMP control packets that are enabled by default and need to be rate-limited. However, when IGMP snooping and IP multicast routing are disabled, the packets are treated like data packets, and no policers are assigned to them.

The switch uses policing to accomplish control-plane security by either dropping or rate-limiting Layer 2 control packets. If a Layer 2 protocol is enabled on a UNI port or tunneled on the switch, those protocol packets are rate-limited; otherwise control packets are dropped.

By default, some protocol traffic is dropped by the CPU, and some is rate-limited. Table 29-1 shows the default action and the action taken for Layer 2 protocol packets when the feature is enabled or when Layer 2 protocol tunneling is enabled for the protocol. Note that some features cannot be enabled on UNIs, and not all protocols can be tunneled (shown by dashes). If Layer 2 protocol tunneling is enabled for *any* of the supported protocols (CDP, STP, VTP, LACP, PAgP, or UDLD), the switch Layer 2 protocol tunneling protocol uses the rate-limiting policer on every port. If UDLD is enabled on a port or UDLD tunneling is enabled, UDLD packets are rate-limited.

Table 29-1 CPU Protection Actions When Layer 2 Protocol Packets Are Received on a UNI

Protocol	Default	When Feature Is Enabled	When Layer 2 Protocol Tunneling Is Enabled [1]
STP	Dropped	–	Rate-limited
RSVD_STP (reserved IEEE 802.1D addresses)	Dropped	–	
PVST+	Dropped	–	Rate-limited
LACP	Dropped	–	Rate-limited
PAgP	Dropped	–	Rate-limited
802.1x	Dropped	Rate-limited	–
CDP	Dropped	–	Rate-limited
DTP	Dropped	–	–
UDLD	Dropped	Rate-limited	Rate-limited
VTP	Dropped	–	Rate-limited

Table 29-1 CPU Protection Actions When Layer 2 Protocol Packets Are Received on a UNI (continued)

Protocol	Default	When Feature Is Enabled	When Layer 2 Protocol Tunneling Is Enabled [1]
CISCO_L2 (any other Cisco Layer 2 protocols with the MAC address 01:00:0c:cc:cc:cc)	Dropped	–	Rate-limited if CDP, DTP, UDLD, PAGP, or VTP are Layer 2 tunneled
KEEPALIVE (MAC address, SNAP encapsulation, LLC, Org ID, or HDLC packets)	Rate-limited	–	–
SWITCH L2PT	Dropped	Rate-limited when Layer 2 protocol tunneling is enabled for any protocol.	Rate-limited

1. Layer 2 protocol traffic is rate-limited when Layer 2 protocol tunneling is enabled for any protocol on any port.

The switch automatically allocates 27 control-plane security policers for CPU protection. At system bootup, it assigns a policer to each port numbered 0 to 26. The policer assigned to a port determines if the protocol packets arriving on the port are rate-limited or dropped. A policer of 26 means a drop policer and is a global policer; any traffic type shown as 26 on any port is dropped. A policer of a value of 0 to 25 means that a rate-limiting policer is assigned to the port for the protocol. The policers 0 to 23 are logical identifiers for Fast Ethernet ports 1 to 24; policers 24 and 25 refer to Gigabit Ethernet ports 1 and 2, respectively. A policer value of 255 means that no policer is assigned to a protocol.

To see what policer actions are assigned to the protocols on an interface, enter the **show platform policer cpu interface** *interface-id* privileged EXEC command. This example shows the default policer configuration for a UNI. Because the port is Fast Ethernet 1, the identifier for rate-limited protocols is 0; a display for Fast Ethernet port 5 would display an identifier of 4. The *Policer Index* refers to the specific protocol.

```
Switch# show platform policer cpu interface fastethernet 0/1
Policers assigned for CPU protection
=========================================================
 Feature                          Policer        Physical
                                  Index          Policer
=========================================================
Fa0/1
STP                                 1             26
LACP                                2             26
8021X                               3             26
RSVD_STP                            4             26
PVST_PLUS                           5             26
CDP                                 6             26
DTP                                 7             26
UDLD                                8             26
PAGP                                9             26
VTP                                10             26
CISCO_L2                           11             26
KEEPALIVE                          12              0
SWITCH_MAC                         13             26
SWITCH_ROUTER_MAC                  14             26
SWITCH_IGMP                        15              0
SWITCH_L2PT                        16             26
```

This example shows the default policers assigned to NNIs. Most protocols have no policers assigned to NNIs. A value of 255 means that no policer is assigned to the port for the protocol.

```
Switch #show platform policer cpu interface gigabitethernet 0/1
Policers assigned for CPU protection
=========================================================
 Feature                          Policer        Physical
                                  Index          Policer
=========================================================
Gi0/1
STP                                 1            255
LACP                                2            255
8021X                               3            255
RSVD_STP                            4            255
PVST_PLUS                           5            255
CDP                                 6            255
DTP                                 7            255
UDLD                                8            255
PAGP                                9            255
VTP                                10            255
CISCO_L2                           11            255
KEEPALIVE                          12            255
SWITCH_MAC                         13            255
SWITCH_ROUTER_MAC                  14            255
SWITCH_IGMP                        15            255
SWITCH_L2PT                        16            255
```

Configuring Control-Plane Security

CPU policers are pre-allocated. You can configure only the rate-limiting threshold. The configured threshold applies to all protocols and all UNIs.

Note During normal Layer 2 operation, you cannot ping the switch through a UNI. This restriction does not apply to NNIs. See the "Using Ping" section on page 36-10 for ways to enable ping in a test situation.

Beginning in privileged EXEC mode, follow these steps to set the threshold rate for CPU protection:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**policer cpu uni** *rate*	Configure the CPU protection policing threshold rate. The range is from 8000 to 409500 bits per second (bps). The default, if none is configured, is 160000 bps.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show policer cpu uni rate**	Verify the configured CPU policer rate.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default threshold rate, use the **no policer cpu uni** global configuration command.

This example shows how to set the CPU protection threshold to 10000 bps and to verify the configuration.

```
Switch# config t
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# policer cpu uni 10000
Switch(config)# end
Switch# show policer cpu uni rate

CPU UNI port police rate = 10000 bps
```

Monitoring Control-Plane Security

You can monitor control-plane security settings and statistics on the switch or on an interface, and you can clear these statistics at any time. Use the privileged EXEC commands in Table 29-2 to display or clear this information. For more information about the commands, see the command reference for this release.

Table 29-2 Commands for Monitoring Control-Plane Security

Command	Purpose
clear policer cpu uni counters {**classification** \| **drop**}	Clears all control-plane statistics per feature (**classification**) or all statistics maintained by the control-plane policer (**drop**).
debug platform policer cpu uni	Enables debugging of the control-plane policer. This command displays information messages when any changes are made to CPU protection.
show platform policer cpu {**classification** \| **interface** *interface-id*}	Displays control-plane policer information. • **classification**—show classification statistics. • **interface** *interface-id*—show policer indexes for the specified interface.
show policer cpu uni {**drop** [*policer-number*] \| **rate**}	Displays CPU policer information for the switch. • **drop** [*policer-number*]—show the number of dropped frames for all policer numbers or the specified policer number. • **rate**—show the configured threshold rate for CPU policers.



Configuring QoS

This chapter describes how to configure quality of service (QoS) by using the modular QoS command-line interface (CLI), or MQC, commands on the Cisco ME 3400 Ethernet Access switch. With QoS, you can provide preferential treatment to certain types of traffic at the expense of others. When QoS is not configured, the switch offers best-effort service to each packet, regardless of the packet contents or size. It sends the packets without any assurance of reliability, delay bounds, or throughput. MQC provides a comprehensive hierarchical configuration framework for prioritizing or limiting specific streams of traffic.

For more information about Cisco IOS MQC commands, see the "Cisco IOS Quality of Service Solutions Command Reference" at this site:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fqos_r/index.htm

For complete syntax and usage information for the platform-specific commands used in this chapter, see the command reference for this release.

This chapter consists of these sections:

- Understanding QoS, page 30-1
- Configuring QoS, page 30-26
- Displaying QoS Information, page 30-55
- Configuration Examples for Policy Maps, page 30-56

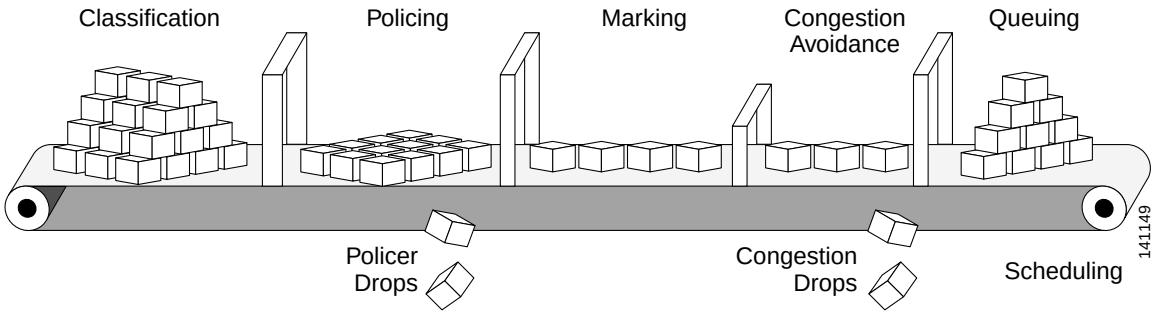
Understanding QoS

Typically, networks operate on a best-effort delivery basis, which means that all traffic has equal priority and an equal chance of being delivered in a timely manner. When congestion occurs, all traffic has an equal chance of being dropped.

When you configure the QoS feature, you can select specific network traffic, prioritize it according to its relative importance, and use traffic-management techniques to provide preferential treatment. Implementing QoS in your network makes network performance more predictable and bandwidth utilization more effective.

Figure 30-1 shows the MQC model.

Figure 30-1 Modular QoS CLI Model



Basic QoS includes these actions.

- Packet classification allows you to organize traffic on the basis of whether or not the traffic matches a specific criteria. When a packet is received, the switch identifies all key packet fields: class of service (CoS), Differentiated Services Code Point (DSCP), or IP precedence. The switch classifies the packet based on this content or based on an access-control list lookup. For more information, see the "Classification" section on page 30-5.
- Packet policing determines whether a packet is in or out of profile by comparing the rate of the incoming traffic to the configured policer. You can control the traffic flow for packets that conform to or exceed the configured policer. For more information, see the "Policing" section on page 30-12.
- Packet prioritization or marking evaluates the classification and policer information to determine the action to take. All packets that belong to a classification can be remarked. When you configure a policer, packets that meet or exceed the permitted bandwidth requirements (bits per second) can be conditionally passed through, dropped, or reclassified. For more information, see the "Marking" section on page 30-16.
- Congestion management uses queuing and scheduling algorithms to queue and sort traffic leaving a port. The switch supports these scheduling and traffic-limiting features: class-based weighted fair queuing (CBWFQ), class-based traffic shaping, port shaping, and class-based priority queuing. You can provide guaranteed bandwidth to a particular class of traffic while still servicing other traffic queues. For more information, see the "Congestion Management and Scheduling" section on page 30-18.
- Queuing on the switch is enhanced with the weighted tail-drop (WTD) algorithm, a congestion-avoidance mechanism. WTD differentiated traffic classes and regulates the queue size (in number of packets) based on the classification. For more information, see the "Congestion Avoidance and Queuing" section on page 30-24.

This section includes information about these topics:

- Modular QoS CLI, page 30-3
- Input and Output Policies, page 30-4
- Classification, page 30-5
- Table Maps, page 30-11
- Policing, page 30-12
- Marking, page 30-16
- Congestion Management and Scheduling, page 30-18
- Congestion Avoidance and Queuing, page 30-24

Modular QoS CLI

Modular QoS CLI (MQC) allows users to create traffic policies and attach these policies to interfaces. A traffic policy contains a traffic class and one or more QoS features. You use a traffic class to classify traffic, and the QoS features in the traffic policy determine how to treat the classified traffic.

Modular QoS CLI configuration includes these steps:

Step 1 Define a traffic class.

Use the **class-map** [**match-all** | **match-any**] *class-map-name* global configuration command to define a traffic class and to enter class-map configuration mode. A traffic class contains three major elements: a name, an instruction on how to evaluate the configured **match** commands (if more than one match command is configured in the class map), and a series of **match** commands

- You name the traffic class in the **class-map** command line to enter class-map configuration mode.
- You can optionally include keywords to evaluate these match commands by entering **class-map match-any** or **class-map match-all**. If you specify **match-any**, the traffic being evaluated must match *one* of the specified criteria. If you specify **match-all**, the traffic being evaluated must match *all* of the specified criteria. A **match-all** class map can contain only one match statement, but a **match-any** class map can contain multiple match statements.



Note If you do not enter **match-all** or **match-any**, the default is to match all.

- You use the **match** class-map configuration commands to specify criteria for classifying packets. If a packet matches the specified criteria, that packet is considered a member of the class and is forwarded according to the QoS specifications set in the traffic policy. Packets that fail to meet any of the matching criteria are classified as members of the default traffic class.

Step 2 Create a traffic policy to associate the traffic class with one or more QoS features.

You use the **policy-map** *policy-map-name* global configuration command to create a traffic policy and to enter policy-map configuration mode. A traffic policy defines the QoS features to associate with the specified traffic class. A traffic policy contains three elements: a name, a traffic class (specified with the **class** policy-map configuration command), and the QoS policies configured in the class.

- You name the traffic policy in the **policy-map** command line to enter policy-map configuration mode.
- In policy-map configuration mode, enter the name of the traffic class used to classify traffic to the specified policy, and enter policy-map class configuration mode.
- In policy-map class configuration mode, you can enter the QoS features to apply to the classified traffic. These include using the **set**, **police**, or **police aggregate** command for input policy maps or the **bandwidth**, **priority**, **queue-limit** or **shape average** command for output policy maps.



Note A packet can match only one traffic class within a traffic policy. If a packet matches more than one traffic class in the traffic policy, the first traffic class defined in the policy is used. To configure more than one match criterion for packets, you can associate multiple traffic classes with a single traffic policy.

Step 3 Attach the traffic policy to an interface.

You use the **service-policy** interface configuration command to attach the policy map to an interface for packets entering or leaving the interface. You must specify whether the traffic policy characteristics should be applied to incoming or outgoing packets. For example, entering the **service-policy output class1** interface configuration command attaches all the characteristics of the traffic policy named *class1* to the specified interface. All packets leaving the specified interface are evaluated according to the criteria specified in the traffic policy named *class1*.

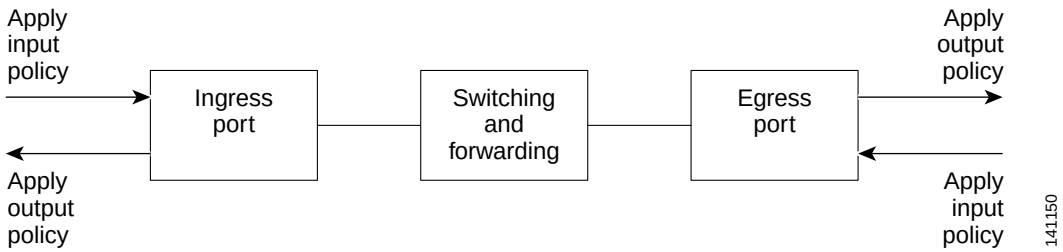
Input and Output Policies

Policy maps are either input policy maps or output policy maps, attached to packets as they enter or leave the switch by service policies applied to interfaces. Input policy maps usually perform policing and marking on received traffic. Policed packets can be dropped or reduced in priority (marked down) if they exceed the maximum permitted rates. Output policy maps perform scheduling and queuing on traffic as it leaves the switch.

Input policies and output policies have the same basic structure; the difference is in the characteristics they regulate. Figure 30-2 shows the relationship of input and output policies.

The maximum number of policy maps that you can configure is 256. You can apply one input policy map and one output policy map to an interface.

Figure 30-2 Input and Output Policy Relationship



Input Policy Maps

Input policy map classification criteria include matching a CoS, a DSCP, or an IP precedence value or matching an ACL. Input policy maps can have any of these actions:

- Setting or marking a CoS, a DSCP, an IP precedence, or a QoS group value
- Individual policing
- Aggregate policing

Only input policies provide matching on access groups, and only output policies provide matching on QoS groups. You can assign a QoS group number in an input policy, and match it in the output policy. The class **class-default** is used in a policy map for any traffic that does not explicitly match any other class in the policy map. Input policy maps do not support queuing and scheduling commands, such as **bandwidth**, **queue-limit**, **priority**, and **shape average**.

You can configure a maximum of 32 total classes in an input policy.

Output Policy Maps

Output policy map classification criteria include matching a CoS, a DSCP, an IP precedence, or a QoS group values. Output policy maps can have any of these actions:

- Queuing (**queue-limit**)
- Scheduling (**bandwidth**, **priority**, and **shape average**)

Output policy maps do not support matching of access groups. You can use QoS groups as an alternative by matching the appropriate access group in the input policy map and setting a QoS group. In the output policy map, you can then match the QoS group. See the "Classification Based on QoS Groups" section on page 30-10 for more information.

Output policies do not support marking or policing (except in the case of priority with policing). There is no egress packet marking on the switch (no **set** command in an output policy).

The class **class-default** is used in a policy map for any traffic that does not explicitly match any other class in the policy map. There can be a maximum of four classes in the output policy map (including class-default) because egress ports have a maximum of four queues.

An output policy map attached to an egress port must match only the packets that have already been matched by an input policy map attached to the ingress port for the packet. The switch supports a limit of three unique output policy maps. You can attach an output policy map to any or all ports on the switch.

Classification

Classification distinguishes one kind of traffic from another by examining the fields in the packet header. When a packet is received, the switch examines the header and identifies all key packet fields. A packet can be classified based on an access-control list, or the DSCP, the CoS, or the IP precedence value in the packet. Figure 30-3 has examples of classification information carried in a Layer 2 or a Layer 3 IP packet header, using six bits from the deprecated IP type of service (ToS) field to carry the classification information.

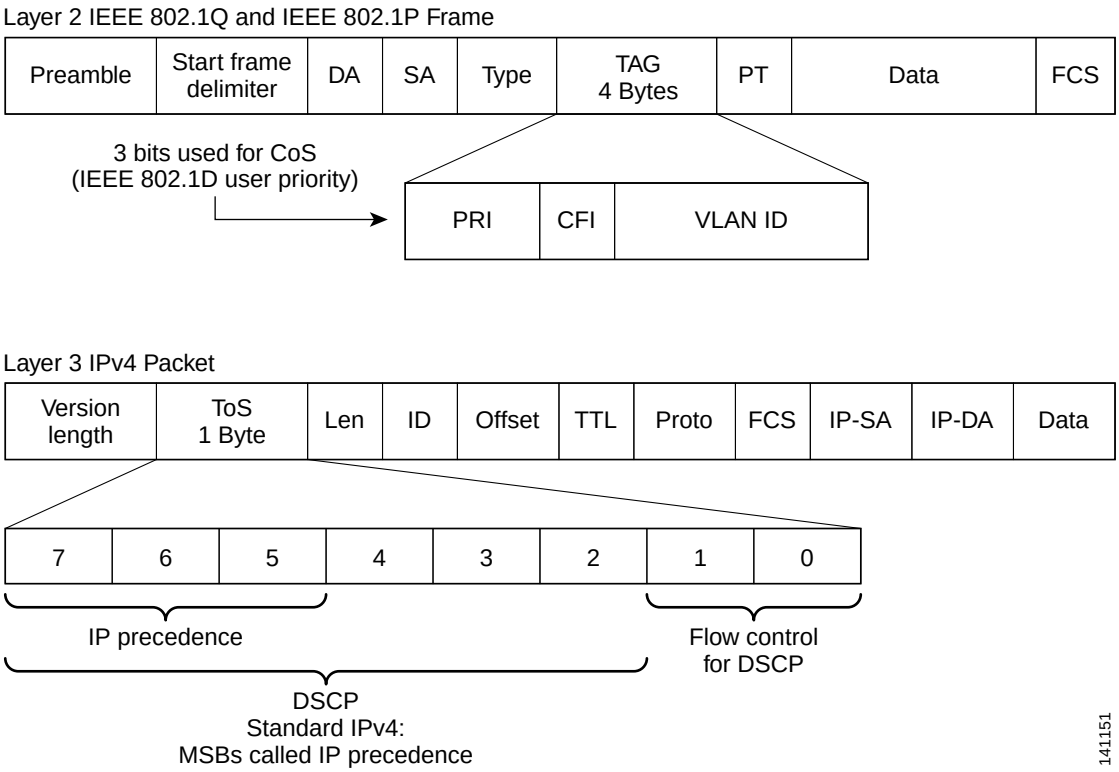
- On ports configured as Layer 2 IEEE 802.1Q trunks, all traffic is in IEEE 802.1Q frames except for traffic in the native VLAN. Layer 2 IEEE 802.1Q frame headers have a 2-byte Tag Control Information field that carries the CoS value in the three most-significant bits, which are called the User Priority bits. Other frame types cannot carry Layer 2 CoS values.

 Layer 2 CoS values range from 0 to 7.

- Layer 3 IP packets can carry either an IP precedence value or a DSCP value. QoS supports the use of either value because DSCP values are backward-compatible with IP precedence values.

 IP precedence values range from 0 to 7. DSCP values range from 0 to 63.

Figure 30-3 QoS Classification Layers in Frames and Packets



These sections contain additional information about classification:

- "Class Maps" section on page 30-6
- "The match Command" section on page 30-7
- "Classification Based on Layer 2 CoS" section on page 30-7
- "Classification Based on IP Precedence" section on page 30-8
- "Classification Based on IP DSCP" section on page 30-8
- "Classification Comparisons" section on page 30-9
- "Classification Based on QoS ACLs" section on page 30-10
- "Classification Based on QoS Groups" section on page 30-10

Class Maps

As explained previously, you use an MQC class map to name a specific traffic flow (or class) and to isolate it from all other traffic. A class map defines the criteria used to match against a specific traffic flow to further classify it. If you have more than one type of traffic that you want to classify, you can create another class map and use a different name. When you enter the **class-map** command with a class-map name, the switch enters the class-map configuration mode. In this mode, you define the match criterion for the traffic by using the **match** class-map configuration command. After a packet is matched against the class-map criteria, it is acted on by the associated action specified in a policy map.

You can match more than one criterion for classification. You can also create a class map that requires that all matching criteria in the class map be in the packet header by using the **class map match-all** *class-map name* global configuration command to enter class map configuration mode. A class map with no match condition has a default of match all. You can configure only one match entry in a **match-all** class map.

Note If you do not enter **match-all** or **match-any**, the default is to match all. A match-all class map cannot have more than one classification criterion (match statement).

You can use the **class map match-any** *class-map name* global configuration command to define a classification with any of the listed criteria.

The match Command

To configure the type of content used to classify packets, you use the **match** class-map configuration command to specify the classification criteria. If a packet matches the configured criteria, it belongs to a specific class and is forwarded according to the specified policy. For example, you can use the **match** command with CoS, IP DSCP, and IP precedence values. These values are referred to as *markings* on a packet. You can also match an access group or a QoS group.

- For an input policy map, you cannot configure an IP classification (**match ip dscp**, **match ip precedence**, **match ip acl**) and a non-IP classification (**match cos** or **match mac acl**) in the same policy map or class map.
- When an input policy map with only Layer 2 classification is attached to a routed port or a switch port containing a routed switch virtual interface (SVI), the service policy acts only on switching eligible traffic and not on routing eligible traffic.
- On an IEEE 802.1Q tunnel port, you can use only an input policy map with Layer 2 classification based on MAC ACLs to classify traffic. Input policy maps with Layer 3 classification or match Layer 2 CoS classification are not supported on tunnel ports.
- In an output policy map, no two class maps can have the same classification criteria, that is, the same match qualifiers and values.

This example shows how to create a class map *example* to define a class that matches any of the listed criteria. In this example, if a packet is received with the DSCP equal to 32 or a 40, the packet is identified (classified) by the class map.

```
Switch(config)# class-map match-any example
Switch(config-cmap)# match ip dscp 32
Switch(config-cmap)# match ip dscp 40
Switch(config-cmap)# exit
```

Classification Based on Layer 2 CoS

You can use the **match** command to classify Layer 2 traffic based on the CoS value, which ranges from 0 to 7.

Note A **match cos** command is supported only on Layer 2 IEEE 802.1Q trunk ports.

This example shows how to create a class map to match a CoS value of 5:

```
Switch(config)# class-map premium
Switch(config-cmap)# match cos 5
Switch(config-cmap)# exit
```

Classification Based on IP Precedence

You can classify IPv4 traffic based on the packet IP precedence values, which range from 0 to 7.

This example shows how to create a class map to match an IP precedence value of 4:

```
Switch(config)# class-map sample
Switch(config-cmap)# match ip precedence 4
Switch(config-cmap)# exit
```

Classification Based on IP DSCP

When you classify IPv4 traffic based on IP DSCP value, and enter the **match ip dscp** class-map configuration command, you have several classification options:

- Entering a specific DSCP value (0 to 63).
- Using the Default service, which corresponds to an IP precedence and DSCP value of 0. The default per-hop behavior (PHB) is usually best-effort service.
- Using Assured Forwarding (AF) by entering the binary representation of the DSCP value. AF sets the relative probability that a specific class of packets is forwarded when congestion occurs and the traffic does not exceed the maximum permitted rate. AF *per-hop behavior* provides delivery of IP packets in four different AF classes: AF11-13 (the highest), AF21-23, AF31-33, and AF41-43 (the lowest). Each AF class could be allocated a specific amount of buffer space and drop probabilities, specified by the binary form of the DSCP number. When congestion occurs, the drop precedence of a packet determines the relative importance of the packet within the class. An AF41 provides the best probability of a packet being forwarded from one end of the network to the other.
- Entering Class Selector (CS) service values of 1 to 7, corresponding to IP precedence bits in the ToS field of the packet.
- Using Expedited Forwarding (EF) to specify a low-latency path. This corresponds to a DSCP value of 46. EF services use priority queuing to preempt lower priority traffic classes.

This display shows the available classification options:

```
Switch(config-cmap)# match ip dscp ?
  <0-63>   Differentiated services codepoint value
  af11     Match packets with AF11 dscp (001010)
  af12     Match packets with AF12 dscp (001100)
  af13     Match packets with AF13 dscp (001110)
  af21     Match packets with AF21 dscp (010010)
  af22     Match packets with AF22 dscp (010100)
  af23     Match packets with AF23 dscp (010110)
  af31     Match packets with AF31 dscp (011010)
  af32     Match packets with AF32 dscp (011100)
  af33     Match packets with AF33 dscp (011110)
  af41     Match packets with AF41 dscp (100010)
  af42     Match packets with AF42 dscp (100100)
  af43     Match packets with AF43 dscp (100110)
  cs1      Match packets with CS1(precedence 1) dscp (001000)
  cs2      Match packets with CS2(precedence 2) dscp (010000)
  cs3      Match packets with CS3(precedence 3) dscp (011000)
  cs4      Match packets with CS4(precedence 4) dscp (100000)
  cs5      Match packets with CS5(precedence 5) dscp (101000)
```

```
cs6      Match packets with CS6(precedence 6) dscp (110000)
cs7      Match packets with CS7(precedence 7) dscp (111000)
default  Match packets with default dscp (000000)
ef       Match packets with EF dscp (101110)
```

For more information on DSCP prioritization, see RFC-2597 (AF per-hop behavior), RFC-2598 (EF), or RFC-2475 (DSCP).

Classification Comparisons

Table 30-1 shows suggested IP DSCP, IP precedence, and CoS values for typical traffic types:

Table 30-1 Typical Traffic Classifications

Traffic Type	DSCP per-hop	DSCP (decimal)	IP Precedence	CoS
Voice-bearer—traffic in a priority queue or the queue with the highest service weight and lowest drop priority.	EF	46	5	5
Voice control—signalling traffic, related to call setup, from a voice gateway or voice application server.	AF31	26	3	3
Video conferencing—in most networks, video conferencing over IP has similar loss, delay, and delay variation requirements as voice over IP traffic.	AF41	34	4	4
Streaming video—relatively high bandwidth applications with a high tolerance for loss, delay, and delay variation. Usually considered more important than regular background applications such as e-mail and web browsing	AF13	14	1	1
Mission critical date (gold data)—delay-sensitive applications critical to the operation of an enterprise.				
Level 1	AF21	18	2	2
Level 2	AF22	20	2	2
Level 3	AF23	22	2	2
Less critical data (silver data)—noncritical, but relatively important data.				
Level 1	AF11	10	1	1
Level 2	AF12	12	1	1
Level 3	AF13	14	1	1
Best-effort data (bronze data)—other traffic, including all noninteractive traffic, regardless of importance.	Default	0	0	0
Less than best-effort data—noncritical, bandwidth-intensive data traffic given the least preference. This is the first traffic type to be dropped.				
Level 1		2	0	0
Level 2		4	0	0
Level 3		6	0	0

Classification Based on QoS ACLs

Packets can also be classified in input policy maps based on an ACL lookup. The ACL classification is communicated to an output policy by assigning a QoS group or number in the input policy map. To classify based on ACL lookup, you first create an IP or MAC ACL. Configure a class map and use the **match access-group** {*acl-number* | *acl name*} class-map configuration command, and attach the class map to a policy map.

Note You cannot configure **match access-group** for an output policy map.

You can use IP standard, IP extended, or Layer 2 MAC ACLs to define a group of packets with the same characteristics (a class). You use the **access-list** global configuration command to configure IP ACLS to classify IP traffic based on Layer 3 and Layer 4 parameters. You use the **mac access-list extended** global configuration command to configure Layer 2 MAC ACLs to classify IP and non-IP traffic based on Layer 2 parameters.

Note You cannot match IP fragments against configured IP extended ACLs to enforce QoS. IP fragments are sent as best-effort. IP fragments are denoted by fields in the IP header.

You can use only ACLs with a permit action in a **match access-group** command. ACLs with a deny action are never matched in a QoS policy.

Note Only one access-group is supported per class for an input policy map.

Classification Based on QoS Groups

A QoS group is an internal label used by the switch to identify packets as a members of a specific class. The label is not part of the packet header and is restricted to the switch that sets the label. QoS groups provide a way to tag a packet for subsequent QoS action without explicitly marking (changing) the packet. You can then communicate an ACL match from an input policy map to an output policy map.

A QoS group is identified at ingress and used at egress; it is assigned in an input policy to identify packets in an output policy. See Figure 30-3. QoS groups help aggregate different classes of input traffic for a specific action in an output policy.

Figure 30-4 QoS Groups



You can use QoS groups to aggregate multiple input streams across input classes and policy maps for the same QoS treatment on the egress port. Assign the same QoS group number in the input policy map to all streams that require the same egress treatment, and match to the QoS group number in the output policy map to specify the required queuing and scheduling actions.

You can also use QoS groups to identify traffic entering a particular interface if the traffic must be treated differently at the output based on the input interface.

To communicate an ACL classification to an output policy, you assign a QoS number to specify packets at ingress. This example identifies specific packets as part of QoS group 1 for later processing in an output policy:

```
Switch(config)# policy-map in-gold-policy
Switch(config-pmap)# class in-class1
Switch(config-pmap-c)# set qos-group 1
Switch(config-cmap-c)# exit
Switch(config-cmap)# exit
```

You use the **set qos-group** command only in an input policy. The assigned QoS group identification is subsequently used in an output policy with no mark or change to the packet. You use the **match qos-group** in the output policy.

Note You cannot configure **match qos-group** for an input policy map.

This example creates an output policy to match the QoS group created in the input policy map *in-gold-policy*. Traffic internally tagged as *qos-group 1* is identified and processed by the output policy.

```
Switch(config)# class-map out-class1
Switch(config-cmap)# match qos-group 1
Switch(config-cmap)# exit
```

The switch supports a maximum of 16 QoS groups.

Table Maps

You can use table maps to manage a large number of traffic flows with a single command. You can specify table maps in **set** commands and use them as mark-down mapping for the policers. You can also use table maps to map an incoming QoS marking to a replacement marking without having to configure a large number of explicit matches and sets. Table maps are used only in input policy maps.

Table maps can be used to:

- correlate specific CoS, DSCP, or IP precedence values to specific CoS, DSCP, or IP precedence values
- mark down a CoS, DSCP, or IP precedence value
- assign defaults for unmapped values

A table map includes one of these default actions:

- **default** *default-value*—applies a specific default value (0 to 63) for all unmapped values
- **default copy**—maps all unmapped values to the equivalent value in another qualifier
- **default ignore**—makes no changes for unmapped values

This example creates a table to map specific CoS values to DSCP values. The default command maps all unmapped CoS values to a DSCP value of 63.

```
Switch(config)# table-map cos-dscp-tablemap
Switch(config-tablemap)# map from 5 to 46
Switch(config-tablemap)# map from 6 to 56
Switch(config-tablemap)# map from 7 to 57
Switch(config-tablemap)# default 63
Switch(config-tablemap)# exit
```

The switch supports a maximum of 256 unique table maps. You can enter up to 64 different **map from–to** entries in a table map. These table maps are supported on the switch:

- DSCP to CoS
- DSCP to precedence
- DSCP to DSCP
- CoS to DSCP
- CoS to precedence
- CoS to CoS
- Precedence to CoS
- Precedence to DSCP
- Precedence to precedence

Table maps modify only one parameter (CoS, IP precedence, or DSCP, whichever is configured) and are only effective when configured with a **set** command in a policy map or with an **exceed-action** command in a police function.

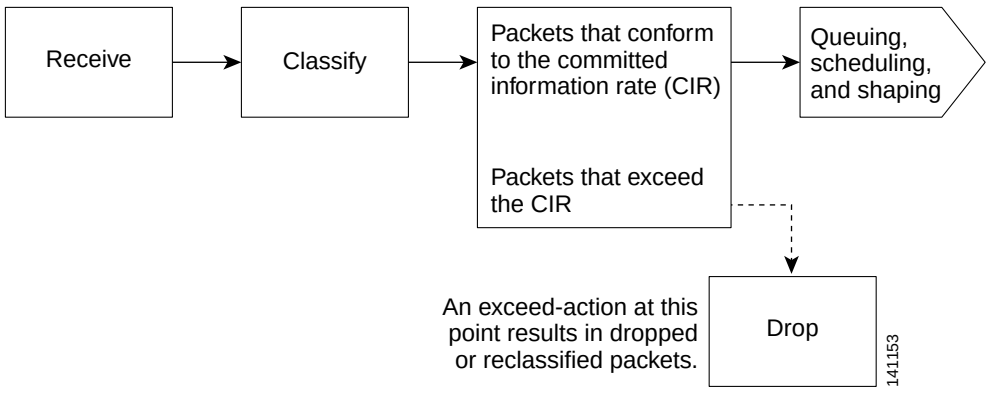
Table maps are not supported in output policy maps. For more information, set the "Configuring Table Maps" section on page 30-33.

Policing

After a packet is classified, you can use policing as shown in Figure 30-5 to regulate the class of traffic. The policing function limits the amount of bandwidth available to a specific traffic flow or prevents a traffic type from using excessive bandwidth and system resources. A policer identifies a packet as in or out of profile by comparing the rate of the inbound traffic to the configuration profile of the policer and traffic class. Packets that exceed the permitted average rate or burst rate are *out of profile* or *nonconforming*. These packets are dropped or modified (marked for further processing), depending on the policer configuration.

Policing is used primarily on receiving interfaces. You can attach a policy map with a policer only in an input service policy. The only policing allowed in an output policy map is in priority classes. See the "Unconditional Priority Policing" section on page 30-15.

Figure 30-5 Policing of Classified Packets



These sections describe the types of policing supported on the switch:

- Individual Policing, page 30-13
- Aggregate Policing, page 30-14
- Unconditional Priority Policing, page 30-15

Individual Policing

Individual policing applies only to input policy maps. In policy-map configuration mode, you enter the **class** command followed by class-map name, and enter policy-map class configuration mode.

Note Individual policing is not supported for the class **class-default**.

You use the **police** policy-map class configuration command to define the policer, the committed rate limitations of the traffic, committed burst size limitations of the traffic, and the action to take for traffic that is below the limits (**conform-action**) and above the limits (**exceed-action**). If you do not specify burst size (bc), the system calculates an appropriate burst size value. The calculated value is appropriate for most applications. For more information, see the "Attaching a Traffic Policy to an Interface" section on page 30-35.

To make the policy map effective, you attach it to a physical port by using the **service-policy input** interface configuration command. Policing is done only on received traffic, so you can only attach a policer to an input service policy.

This is an example of basic policing for all traffic received with a CoS of 4. The first value following the **police** command limits the average traffic rate to 10, 000,000 bits per second (bps); the second value represents the additional burst size (10 kilobytes). The policy is assigned to Fast Ethernet port 1.

```
Switch(config)# class-map video-class
Switch(config-cmap)# match cos 4
Switch(config-cmap)# exit
Switch(config)# policy-map video-policy
Switch(config-pmap)# class video-class
Switch(config-pmap-c)# police 10000000 10000
Switch(config-pmap-c-police)# exit
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface fastethernet0/1
Switch(config-if)# service-policy input video-policy
Switch(config-if)# exit
```

You can use the **conform-action** and **exceed-action** policy-map class configuration commands or the **conform-action** and **exceed-action** policy-map class police configuration commands to specify the action to be taken when the packet conforms to or exceeds the specified traffic rate.

Note You cannot configure marking simultaneously in both **conform-action** and **exceed-action** for the same policer instance.

Conform actions are to send the packet without modifications, to set a QoS group value for classification at the egress, or to set a new CoS, DSCP, or IP precedence value. Exceed actions are to drop the packet or to set a new CoS, DSCP, or IP precedence to a value defined in a table map and then send the packet. Table maps list specific traffic attributes and map (or convert) them to other attributes.

After you create a table map, you configure a policy-map policer to use the table map.

- You can configure only one **exceed-action** police markdown table map of each type (CoS, DSCP, or IP precedence) on the switch. You can reference that table map in multiple policers.
- When you use a table map in an input policy map, the protocol type for the **from**–action in the table map must be the same as the protocol type of the associated classification. For example, if a class map represents IP classification, the **from**–type action in the table map must be either **dscp** or **precedence**. If the class map represents a non-IP classification, the **from**–type action in the table map must be **cos**.

To configure multiple actions in a class, you can enter multiple conform or exceed action entries in policy-map class police configuration mode, as in this example:

```
Switch(config)# policy-map map1
Switch(config-pmap)# class class1
Switch(config-pmap-c)# police 100000 500000
Switch(config-pmap-c-police)# conform-action set-cos-transmit 4
Switch(config-pmap-c-police)# conform-action set-dscp-transmit 48
Switch(config-pmap-c-police)# exceed-action drop
Switch(config-pmap-c-police)# exit
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
```

Aggregate Policing

Aggregate policing applies only to input policy maps. An aggregate policer differs from an individual policer because it is shared by multiple traffic classes within a policy map.

Note Aggregate policing is not supported for the class **class-default**.

You can use the **policer aggregate** global configuration command to set a policer for all traffic received or sent on a physical interface. When you configure an aggregate policer, you can configure specific burst sizes and conform and exceed actions. If you do not specify burst size (**bc**), the system calculates an appropriate burst size value. The calculated value is appropriate for most applications.

Note You cannot configure marking simultaneously in both **conform-action** and **exceed-action** for the same policer instance.

Conform actions are to send the packet without modifications, to set a QoS group value for classification at the egress, or to set a new CoS, DSCP, or IP precedence value. Exceed actions are to drop the packet or to set a new CoS, DSCP, or IP precedence to a value defined in a table map and then send the packet. Table maps list specific traffic attributes and map (or convert) them to other attributes.

After you create a table map, you configure a policy-map policer to use the table map.

- You can configure only one **exceed-action** police markdown table map of each type (CoS, DSCP, or IP precedence) on the switch. You can reference that table map in multiple policers.
- When you use a table map in an input policy map, the protocol type for the **from**–action in the table map must be the same as the protocol type of the associated classification. For example, if a class map represents IP classification, the **from**–type action in the table map must be either **dscp** or **precedence**. If the class map represents a non-IP classification, the **from**–type action in the table map must be **cos**.

After you configure the aggregate policer, you create a policy map and an associated class map, associate the policy map with the aggregate policer, and apply the service policy to a port.

Note Only one policy map can use any specific aggregate policer. Aggregate policing cannot be used to aggregate traffic streams across multiple interfaces. It can be used only to aggregate traffic streams across multiple classes in a policy map attached to an interface.

After you configure the policy map and policing actions, attach the policy to an ingress port by using the **service-policy** interface configuration command.

The class maps in this example refer to access lists.

```
Switch(config)# policer aggregate agg1 cir 23000 bc 10000 conform-action set-dscp-transmit
46 exceed-action drop
Switch(config)# class-map testclass
Switch(config-cmap)# match access-group 1
Switch(config-cmap)# exit
Switch(config)# class-map videoclass
Switch(config-cmap)# match access-group 2
Switch(config-cmap)# exit
Switch(config)# policy-map testexample
Switch(config-pmap)# class testclass
Switch(config-pmap-c)# police aggregate agg1
Switch(config-pmap-c)# exit
Switch(config-pmap)# class video-class
Switch(config-pmap-c)# police aggregate agg1
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface fastethernet0/1
Switch(config-if)# service-policy input testexample
Switch(config-if)# exit
```

For configuration information, see the "Configuring Input Policy Maps with Aggregate Policing" section on page 30-39.

Unconditional Priority Policing

Priority policing applies only to output policy maps. You can use the **priority** policy-map class configuration command in an output policy map to designate a low-latency path, or class-based priority queuing, for a specific traffic class. With strict priority queuing, the packets in the priority queue are scheduled and sent until the queue is empty, at the expense of other queues. Excessive use of high-priority queuing can create congestion for lower priority traffic.

To eliminate this congestion, you can use the priority with police feature (priority policing) to reduce the bandwidth used by the priority queue and allocate traffic rates on other queues. Priority with police is the only form of policing supported in output policy maps. See also the "Configuring Output Policy Maps with Class-Based Priority Queuing" section on page 30-48.

Note You cannot configure a policer committed burst size for an unconditional priority policer. Any configured burst size is ignored.

This example shows how to use the **priority** with **police** commands to configure *out-class1* as the priority queue, with traffic going to the queue limited to 20,000,000 bps so that the priority queue never uses more than that. Traffic above that rate is dropped. This allows other traffic queues to receive some port bandwidth, in this case a minimum bandwidth guarantee of 500,000 and 200,000 kbps. The class **class-default** queue gets the remaining port bandwidth.

```
Switch(config)# policy-map policy1
Switch(config-pmap)# class out-class1
Switch(config-pmap-c)# priority
Switch(config-pmap-c)# police 200000000
Switch(config-pmap-c)# exit
Switch(config-pmap)# class out-class2
Switch(config-pmap-c)# bandwidth 500000
Switch(config-pmap-c)# exit
Switch(config-pmap)# class out-class3
Switch(config-pmap-c)# bandwidth 200000
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# service-policy output policy1
Switch(config-if)# exit
```

Marking

You can use packet marking in input policy maps to set or modify the attributes for traffic belonging to a specific class. After network traffic is organized into classes, you use marking to identify certain traffic types for unique handling. For example, you can change the CoS value in a class or set IP DSCP or IP precedence values for a specific type of traffic. These new values are then used to determine how the traffic should be treated. You can also use marking to assign traffic to a QoS group within the switch.

Traffic marking is typically performed on a specific traffic type at the ingress port. The marking action can cause the CoS, DSCP, or precedence bits to be rewritten or left unchanged, depending on the configuration. This can increase or decrease the priority of a packet through the system. The marking function can use information from the policing function or directly from the classification function.

You can specify and mark traffic by using the **set** commands in a policy map for all supported QoS markings (CoS, IP DSCP, IP precedence, and QoS groups). A **set** command unconditionally *marks* the packets that match a specific class. You then attach the policy map to an interface as an input policy map.

Note When you configure a **set qos-group** command in an input policy map, you cannot configure other **set** operations in the same class (such as **set dscp**, **set cos**, or **set precedence**).

You can also mark traffic by using the **set** command with table maps.Table maps list specific traffic attributes and maps (or converts) them to another attribute. A table map establishes a to-from relationship for the attribute and defines the change to be made.

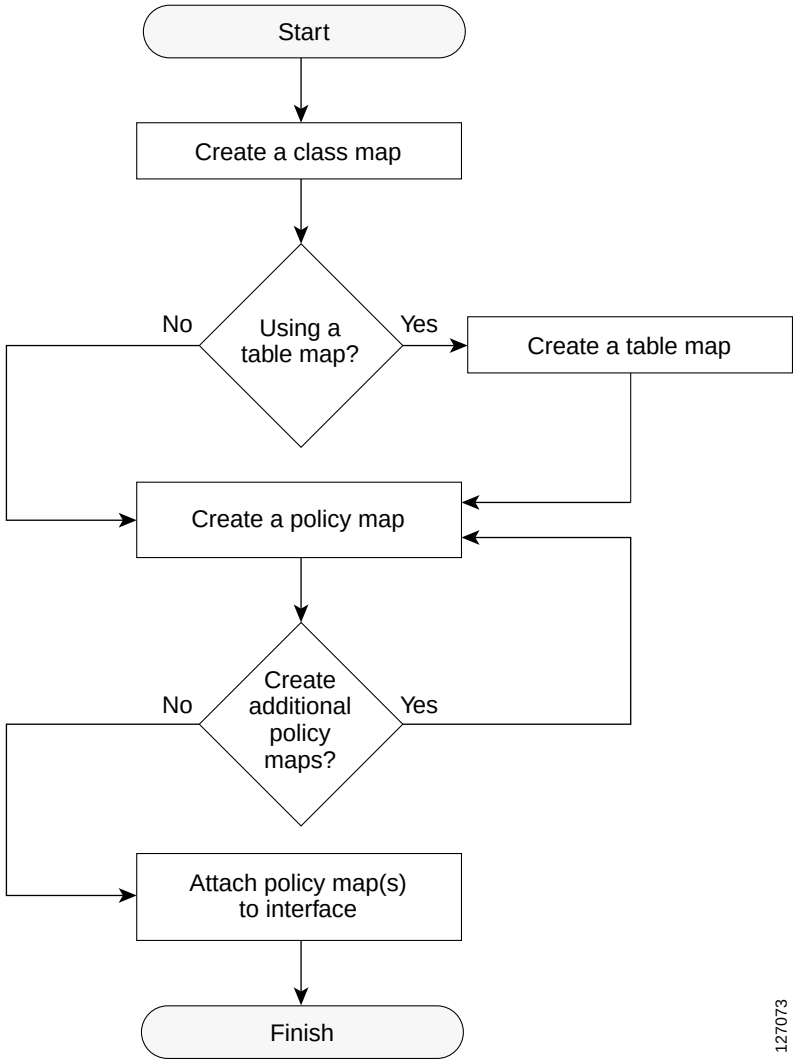
Note When you use a table map in an input policy map, the protocol type of the **from**-type action in the table map must be the same as the protocol type of the associated classification. For example, if a class map represents an IP classification, the **from**-type action in the table map must be either **dscp** or **precedence**. If the class map represents a non-IP classification, the **from**-type action in the table map must be **cos**.

After you create a table map, you configure a policy map to use the table map. See the "Congestion Management and Scheduling" section on page 30-18.

Note You configure only one **set** action with a table map in a class. You cannot configure any other **set** action in the same class as a **set** action with table map.

Figure 30-6 shows the procedures for marking traffic.

Figure 30-6 Flow Chart for Marking Traffic



This example uses a policy map to remark a packet. The first marking (the **set** command) applies to the QoS default class map that matches all traffic not matched by class *AF31-AF33* and sets all traffic to an IP DSCP value of 1. The second marking sets the traffic in classes AF31 to AF33 to an IP DSCP of 3.

```
Switch(config)# policy-map Example
Switch(config-pmap)# class class-default
Switch(config-pmap-c)# set ip dscp 1
Switch(config-pmap-c)# exit
Switch(config-pmap)# class AF31-AF33
Switch(config-pmap-c)# set ip dscp 3
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface fastethernet0/1
Switch(config-if)# service-policy input Example
Switch(config-if)# exit
```

Congestion Management and Scheduling

MQC provides several related mechanisms to control outgoing traffic flow. These mechanisms are implemented in output policy maps to control output traffic queues. The scheduling stage holds packets until the appropriate time to send them to one of the four traffic queues. Queuing assigns a packet to a particular queue based on the packet class, and is enhanced by the WTD algorithm for congestion avoidance. You can use different scheduling mechanisms to provide a guaranteed bandwidth to a particular class of traffic while also serving other traffic in a fair way. You can limit the maximum bandwidth that can be consumed by a particular class of traffic and ensure that delay-sensitive traffic in a low-latency queue is sent before traffic in other queues.

The switch supports these scheduling mechanisms:

- Traffic traffic shaping

 You use the **shape average** policy map class configuration command to specify that a class of traffic should have a maximum permitted average rate. You specify the maximum rate in bits per second.

- Class-based-weighted-fair-queuing (CBWFQ)

 You can use the **bandwidth** policy-map class configuration command to control the bandwidth allocated to a specific class. Minimum bandwidth can be specified as a bit rate or a percentage of total bandwidth or of remaining bandwidth.

- Priority queuing or class-based priority queuing

 You use the **priority** policy-map class configuration command to specify the priority of a type of traffic over other types of traffic. You can specify strict priority for the high-priority traffic and allocate any excess bandwidth to other traffic queues, or specify priority with unconditional policing of high-priority traffic and allocate the known remaining bandwidth among the other traffic queues.

 – To configure strict priority, use only the **priority** policy-map class configuration command to configure the priority queue. Use the **bandwidth remaining percent** policy-map class configuration command for the other traffic classes to allocate the excess bandwidth in the desired ratios.

 – To configure priority with unconditional policing, configure the priority queue by using the **priority** policy-map class configuration command and the **police** policy-map class configuration command to unconditionally rate-limit the priority queue. In this case, you can configure the other traffic classes with **bandwidth** or **shape average**, depending on requirements.

These sections contain additional information about scheduling:

- Traffic Shaping, page 30-19
- Class-Based Weighted Fair Queuing, page 30-21
- Priority Queuing, page 30-22

Traffic Shaping

Traffic shaping is a traffic-control mechanism similar to traffic policing. While traffic policing is used in input policy maps, traffic shaping occurs as traffic leaves an interface. The switch can apply class-based shaping to classes of traffic leaving an interface and port shaping to all traffic leaving an interface. Configuring a queue for traffic shaping sets the maximum bandwidth or peak information rate (PIR) of the queue.

Note You cannot configure traffic shaping (**shape average**) and CBWFQ (**bandwidth**) or priority queuing (**priority**) for the same class in an output policy map. You cannot configure traffic shaping for a traffic class when strict priority (priority without police) is configured for another class within the output policy-map.

Class-Based Shaping

Class-based shaping uses the **shape average** policy-map class configuration command to limit the rate of data transmission as the number of bits per second to be used for the committed information rate for a class of traffic. The switch supports separate queues for three classes of traffic. The fourth queue is always the default queue for class **class-default**, unclassified traffic.

Note In the Cisco ME switch, configuring traffic shaping also automatically sets the minimum bandwidth guarantee or committed information rate (CIR) of the queue to the same value as the PIR.

This example shows how to configure traffic shaping for outgoing traffic on a Fast Ethernet port so that *outclass1*, *outclass2*, and *outclass3* get a maximum of 50, 20, and 10 Mbps, respectively, of the available port bandwidth. The class **class-default** gets the remaining bandwidth.

```
Switch(config)# policy-map out-policy
Switch(config-pmap)# class classout1
Switch(config-pmap-c)# shape average 50000000
Switch(config-pmap-c)# exit
Switch(config-pmap)# class classout2
Switch(config-pmap-c)# shape average 20000000
Switch(config-pmap-c)# exit
Switch(config-pmap)# class classout3
Switch(config-pmap-c)# shape average 10000000
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface fastethernet 0/1
Switch(config-if)# service-policy output out-policy
Switch(config-if)# exit
```

Port Shaping

To configure port shaping (a transmit port shaper), create a policy map that contains only a default class, and use the **shape average** command to specify the maximum bandwidth for a port.

This example shows how to configure a policy map that shapes a port to 90 Mbps, allocated according to the *out-policy* policy map configured in the previous example. The **service-policy** policy map class command is used to create a child policy to the parent:

```
Switch(config)# policy-map out-policy-parent
Switch(config-pmap)# class class-default
Switch(config-pmap-c)# shape average 90000000
Switch(config-pmap-c)# service-policy out-policy
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface fastethernet0/1
Switch(config-if)# service-policy output out-policy-parent
Switch(config-if)# exit
```

Parent-Child Hierarchy

The switch supports two policy levels: a *parent* level and a *child* level. The QoS parent-child structure is used for specific purposes where a child policy is referenced in a parent policy to provide additional control of a specific traffic type.

The first policy level, the parent level, is used for port shaping, and you can specific only one class of type **class-default** within the policy. This is an example of a parent-level policy map:

```
Switch(config)# policy-map parent
Switch(config-pmap)# class class-default
Switch(config-pmap-c)# shape average 50000000
Switch(config-pmap-c)# exit
```

The second policy level, the *child* level, is used to control a specific traffic stream or class, as in this example:

```
Switch(config)# policy-map child
Switch(config-pmap)# class class1
Switch(config-pmap-c)# priority
Switch(config-pmap-c)# exit
```

Note The total of the minimum bandwidth guarantees (CIR) for each queue of the child policy cannot exceed the total port-shape rate.

This is an example of a parent-child configuration:

```
Switch(config)# policy-map parent
Switch(config-pmap)# class class-default
Switch(config-pmap-c)# shape average 50000000
Switch(config-pmap-c)# service-policy child
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface fastethernet0/1
Switch(config-if)# service-policy output parent
Switch(config-if)# exit
```

Class-Based Weighted Fair Queuing

You can configure class-based weighted fair queuing (CBWFQ) to set the relative precedence of a queue by allocating a portion of the total bandwidth that is available for the port. You use the **bandwidth** policy-map class configuration command to set the output bandwidth for a class of traffic as a rate (kilobits per second), a percentage of total bandwidth, or a percentage of remaining bandwidth.

Note When you configure bandwidth in a policy map, you must configure all rates in the same format, either a configured rate or a percentage. The total of the minimum bandwidth guarantees (CIR) for each queue of the policy cannot exceed the total speed of the parent.

- When you use the **bandwidth** policy-map class configuration command to configure a class of traffic as an absolute rate (kilobits per second) or a percentage of total bandwidth, this represents the minimum bandwidth guarantee (CIR) for that traffic class. This means that the traffic class gets at least the bandwidth indicated by the command, but is not limited to that bandwidth. Any excess bandwidth on the port is allocated to each class in the same ratio in which the CIR rates are configured.

 Note You cannot configure bandwidth as an absolute rate or a percentage of total bandwidth when strict priority (priority without police) is configured for another class in the output policy.

- When you use the **bandwidth** policy-map class configuration command to configure a class of traffic as a percentage of *remaining* bandwidth, this represents the portion of the excess bandwidth of the port that is allocated to the class. This means that the class is allocated bandwidth only if there is excess bandwidth on the port, and if there is no minimum bandwidth guarantee for this traffic class.

 Note You can configure bandwidth as percentage of remaining bandwidth only when strict priority (priority without police) is configured for another class in the output policy map.

For more information, see the "Configuring Output Policy Maps with Class-Based-Weighted-Queuing" section on page 30-44.

Note You cannot configure bandwidth and traffic shaping (**shape average**) or priority queuing (**priority**) for the same class in an output policy map.

This example shows how to set the precedence of output queues by setting bandwidth in kilobits per second. The classes *outclass1*, *outclass2*, and *outclass3* get a minimum of 50000, 20000, and 10000 kbps. The class **class-default** at a minimum gets the remaining bandwidth.

```
Switch(config)# policy-map out-policy
Switch(config-pmap)# class outclass1
Switch(config-pmap-c)# bandwidth 50000
Switch(config-pmap-c)# exit
Switch(config-pmap)# class outclass2
Switch(config-pmap-c)# bandwidth 20000
Switch(config-pmap-c)# exit
Switch(config-pmap)# class outclass3
Switch(config-pmap-c)# bandwidth 10000
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
```

```
Switch(config)# interface fastethernet 0/1
Switch(config-if)# service-policy output out-policy
Switch(config-if)# exit
```

This example shows how to allocate the excess bandwidth among queues by configuring bandwidth for a traffic class as a percentage of remaining bandwidth. The class *outclass1* is given priority queue treatment. The other classes are configured to get percentages of the excess bandwidth if any remains after servicing the priority queue: *outclass2* is configured to get 50 percent, *outclass3* to get 20 percent, and the class **class-default** to get the remaining 30 percent.

```
Switch(config)# policy-map out-policy
Switch(config-pmap)# class outclass1
Switch(config-pmap-c)# priority
Switch(config-pmap-c)# exit
Switch(config-pmap)# class outclass2
Switch(config-pmap-c)# bandwidth remaining percent 50
Switch(config-pmap-c)# exit
Switch(config-pmap)# class outclass3
Switch(config-pmap-c)# bandwidth remaining percent 20
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface fastethernet 0/1
Switch(config-if)# service-policy output out-policy
Switch(config-if)# exit
```

Priority Queuing

You can use the **priority** policy-map class configuration command to ensure that a particular class of traffic is given preferential treatment. With strict priority queuing, the priority queue is constantly serviced. All packets in the queue are scheduled and sent until the queue is empty. Priority queuing allows traffic for the associated class to be sent before packets in other queues are sent.

Note You should exercise care when using the **priority** command. Excessive use of strict priority queuing might cause congestion in other queues.

The switch supports strict priority queuing or priority used with the **police** policy-map command.

- *Strict priority queuing* (priority without police) assigns a traffic class to a low-latency queue to ensure that packets in this class have the lowest possible latency. When this is configured, the priority queue is continually serviced until it is empty, possibly at the expense of packets in other queues.

 Note You cannot configure priority without policing for a traffic class when traffic shaping or CBWFQ are configured for another class in the same output policy map.

- You can use priority with the **police** policy-map command, or *unconditional priority policing*, to reduce the bandwidth used by the priority queue. This is the only form of policing that is supported in output policy maps. Using this combination of commands configures a maximum rate on the priority queue, and you can use the **bandwidth** and **shape average** policy-map commands for other classes to allocate traffic rates on other queues.

Note When priority is configured in an output policy map *without* the **police** command, you can only configure the other queues for sharing by using the **bandwidth remaining percent** policy-map command to allocate excess bandwidth.

Priority queuing has these restrictions:

- You can associate the **priority** command with a single unique class for all attached output polices on the switch.
- You cannot configure priority and any other scheduling action (**shape average** or **bandwidth**) in the same class.

For more information, see the "Configuring Output Policy Maps with Class-Based Priority Queuing" section on page 30-48.

This example shows how to configure the class *out-class1* as a strict priority queue so that all packets in that class are sent before any other class of traffic. Other traffic queues are configured so that *out-class-2* gets 50 percent of the remaining bandwidth and *out-class3* gets 20 percent of the remaining bandwidth. The class **class-default** receives the remaining 30 percent with no guarantees.

```
Switch(config)# policy-map policy1
Switch(config-pmap)# class out-class1
Switch(config-pmap-c)# priority
Switch(config-pmap-c)# exit
Switch(config-pmap)# class out-class2
Switch(config-pmap-c)# bandwidth remaining percent 50
Switch(config-pmap-c)# exit
Switch(config-pmap)# class out-class3
Switch(config-pmap-c)# bandwidth remaining percent 20
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface gigabitethernet 0/1
Switch(config-if)# service-policy output policy1
Switch(config-if)# exit
```

This example shows how to use the **priority** with **police** commands to configure *out-class1* as the priority queue, with traffic going to the queue limited to 20000000 bps so that the priority queue will never use more than that. Traffic above that rate is dropped. The other traffic queues are configured to use 50 and 20 percent of the bandwidth that is left, as in the previous example.

```
Switch(config)# policy-map policy1
Switch(config-pmap)# class out-class1
Switch(config-pmap-c)# priority
Switch(config-pmap-c)# police 200000000
Switch(config-pmap-c)# exit
Switch(config-pmap)# class out-class2
Switch(config-pmap-c)# bandwidth percent 50
Switch(config-pmap-c)# exit
Switch(config-pmap)# class out-class3
Switch(config-pmap-c)# bandwidth percent 20
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface gigabitethernet 0/1
Switch(config-if)# service-policy output policy1
Switch(config-if)# exit
```

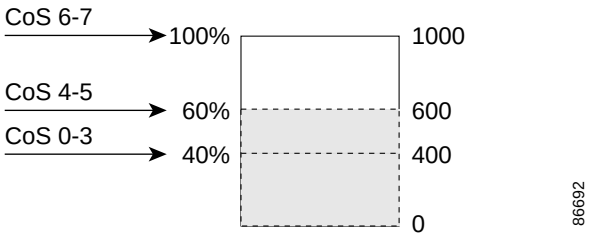
Congestion Avoidance and Queuing

Congestion avoidance uses algorithms such as tail drop to control the number of packets entering the queuing and scheduling stage to avoid congestion and network bottlenecks. The switch uses weighted tail drop (WTD) to manage the queue sizes and provide a drop precedence for traffic classifications. You set the size limits depending on the markings of the packets in the queue. Each packet that travels through the switch can be assigned to a specific queue and threshold. For example, specific DSCP or CoS values can be mapped to a specific egress queue and threshold.

WTD is implemented on traffic queues to manage the queue size and to provide drop precedences for different traffic classifications. As a frame enters a particular queue, WTD uses the packet classification to subject it to different thresholds. If the threshold is exceeded for that classification (the space available in the destination queue is less than the size of the frame), the switch drops the frame.

Figure 30-7 shows an example of WTD operating on a queue of 1000 frames. Three drop percentages are configured: 40 percent (400 frames), 60 percent (600 frames), and 100 percent (1000 frames). These percentages mean that up to 400 frames can be queued at the 40-percent threshold, up to 600 frames at the 60-percent threshold, and up to 1000 frames at the 100-percent threshold.

Figure 30-7 WTD and Queue Operation



In this example, CoS values 6 and 7 have a greater importance than the other CoS values, and they are assigned to the 100-percent drop threshold (queue-full state). CoS values 4 and 5 are assigned to the 60-percent threshold, and CoS values 0 to 3 are assigned to the 40-percent threshold.

If the queue is already filled with 600 frames, and a new frame arrives containing CoS values 4 and 5, the frame is subjected to the 60-percent threshold. When this frame is added to the queue, the threshold would be exceeded, so the switch drops it.

WTD is configured by using the **queue-limit** policy-map class command. The command adjusts the queue size (buffer size) associated with a particular class of traffic. You specify the threshold as the number of packets, where each packet is a fixed unit of 256 bytes. You can specify different queue sizes for different classes of traffic (CoS, DSCP, precedence, or QoS group) in the same queue. Setting a queue limit establishes a drop threshold for the associated traffic when congestion occurs.

Note You cannot configure queue size by using the **queue-limit** policy map class command without first configuring a scheduling action (**bandwidth**, **shape average**, or **priority**).

For more information, see the "Configuring Output Policy Maps with Class-Based-Weighted-Queuing" section on page 30-44.

This example configures *class A* to match DCSP values and a policy map, *PM1*. The DSCP values of 30 and 50 are mapped to unique thresholds (32 and 64, respectively). The DSCP values of 40 and 60 are mapped to the maximum threshold of 112 packets.

```
Switch(config)# class-map match-any classA
Switch(config-cmap)# match ip dscp 30 40 50 60
Switch(config-cmap)# exit
Switch(config)# policy-map PM1
Switch(config-pmap)# class classA
Switch(config-pmap-c)# bandwidth percent 50
Switch(config-pmap-c)# queue-limit 112
Switch(config-pmap-c)# queue-limit dscp 30 32
Switch(config-pmap-c)# queue-limit dscp 50 64
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface gigabitethernet 0/1
Switch(config-if)# service-policy output PM1
Switch(config-if)# exit
```

Note When you configure a queue limit for a class in an output policy map, all other output policy maps must use the same format of qualifier type and qualifier value. Only the queue-limit threshold values can be different. For example, when you configure *class A* queue limit thresholds for **dscp 30** and **dscp 50** in policy map *PM1*, if you configure *class A* queue limits in policy map *PM2*, you must use **dscp 30** and **dscp 50** as qualifiers. You cannot use **dscp 20** and **dscp 40**. Only the threshold values can be different.

By default, the total amount of buffer space is divided equally among all ports and all queues per port, which is adequate for many applications. You can decrease queue size for latency-sensitive traffic or increase queue size for bursty traffic.

Note When you use the **queue-limit** command to configure queue thresholds for a class, the WTD thresholds must be less than or equal to the queue maximum threshold. A queue size configured with no qualifier must be larger than any queue sizes configured with qualifiers.

Queue bandwidth and queue size (queue limit) are configured separately and are not interdependent. You should consider the type of traffic being sent when you configure bandwidth and queue-limit:

- A large buffer (queue limit) can better accommodate bursty traffic without packet loss, but at the cost of increased latency.
- A small buffer reduces latency but is more appropriate for steady traffic flows than for bursty traffic.
- Very small buffers are typically used to optimize priority queuing. For traffic that is priority queued, the buffer size usually needs to accommodate only a few packets; large buffer sizes that increase latency are not usually necessary. For high-priority latency-sensitive packets, configure a relatively large bandwidth and relatively small queue size.

Note These restrictions apply to WTD qualifiers:

- You cannot configure more than two threshold values for WTD qualifiers (**cos**, **dscp**, **precedence**, **qos-group**) by using the **queue-limit** command. However, there is no limit to the number of qualifiers that you can map to these thresholds. You can configure a third threshold value to set the maximum queue by using the **queue-limit** command with no qualifiers.

- A WTD qualifier in the **queue-limit** command must be the same as at least one **match** qualifier in the associated class map.

This example shows how to configure bandwidth and queue limit so that *out-class1*, *out-class2*, and *out-class3* get a minimum of 50, 20, and 10 percent of the traffic bandwidth, and the remaining traffic (class-default) gets the remaining 20 percent. The corresponding queue-size is set to 48, 32, and 16 (256-byte) packets, respectively:

```
Switch(config)# policy-map out-policy
Switch(config-pmap)# class outclass1
Switch(config-pmap-c)# bandwidth percent 50
Switch(config-pmap-c)# queue-limit 48
Switch(config-pmap-c)# exit
Switch(config-pmap)# class outclass2
Switch(config-pmap-c)# bandwidth percent 20
Switch(config-pmap-c)# queue-limit 32
Switch(config-pmap-c)# exit
Switch(config-pmap)# class outclass3
Switch(config-pmap-c)# bandwidth percent 10
Switch(config-pmap-c)# queue-limit 16
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface gigabitethernet 0/1
Switch(config-if)# service-policy output out-policy
Switch(config-if)# exit
```

Configuring QoS

Before configuring QoS, you must have a thorough understanding of these factors:

- The types of applications used and the traffic patterns on your network.
- Traffic characteristics and needs of your network. Is the traffic bursty? Do you need to reserve bandwidth for voice and video streams?
- Bandwidth requirements and speed of the network.
- Location of congestion points in the network.

These sections describe how to classify, police, and mark incoming traffic, and schedule and queue outgoing traffic. Depending on your network configuration, you must perform one or more of these tasks.

- Default QoS Configuration, page 30-27
- QoS Configuration Guidelines, page 30-27
- Using ACLs to Classify Traffic, page 30-27
- Using Class Maps to Define a Traffic Class, page 30-31
- Configuring Table Maps, page 30-33
- Attaching a Traffic Policy to an Interface, page 30-35
- Configuring Input Policy Maps, page 30-35
- Configuring Output Policy Maps, page 30-43

Default QoS Configuration

There are no policy maps, class maps, table maps, or policers configured. At the egress port, all traffic goes through a single default queue that is given the full operational port bandwidth. The default size of the default queue is 48 (256-byte) packets.

The packets are not modified (the CoS, DSCP, and IP precedence values in the packet are not changed). Traffic is switched in pass-through mode without any rewrites and classified as best effort without any policing.

QoS Configuration Guidelines

Before beginning the QoS configuration, you should be aware of these conditions:

- You can configure QoS only on physical ports.
- On a port configured for QoS, all traffic received through the port is classified, policed, and marked according to the input policy map attached to the port. On a trunk port configured for QoS, traffic in *all* VLANs received through the port is classified, policed, and marked according to the policy map attached to the port.
- If you have EtherChannel ports configured on your switch, you must configure QoS classification, policing, mapping, and queuing on the individual physical ports that comprise the EtherChannel. You must decide whether the QoS configuration should match on all ports in the EtherChannel.
- Control traffic (such as spanning-tree bridge protocol data units [BPDUs] and routing update packets) received by the switch are subject to all ingress QoS processing.
- You are likely to lose data when you change queue settings; therefore, try to make changes when traffic is at a minimum.

See the configuration sections for specific QoS features for more configuration guidelines related to each feature.

Using ACLs to Classify Traffic

You can classify IP traffic by using IP standard or IP extended ACLs. You can classify IP and non-IP traffic by using Layer 2 MAC ACLs. For more information about configuring ACLs, see Chapter 28, “Configuring Network Security with ACLs.”

Follow these guidelines when configuring QoS ACLs:

- You cannot match IP fragments against configured IP extended ACLs to enforce QoS. IP fragments are sent as best-effort. IP fragments are denoted by fields in the IP header.
- The switch supports only one access group per class in an input policy map.
- You cannot configure **match-access** group in an output policy map.

These sections describe how to create QoS ACLs:

- “Creating IP Standard ACLs” section on page 30-28
- “Creating IP Extended ACLs” section on page 30-29
- “Creating Layer 2 MAC ACLs” section on page 30-30

Creating IP Standard ACLs

Beginning in privileged EXEC mode, follow these steps to create an IP standard ACL for IP traffic:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**access-list** *access-list-number* **permit** *source* [*source-wildcard*]	Create an IP standard ACL, repeating the command as many times as necessary. • For *access-list-number*, enter the access list number. The range is 1 to 99 and 1300 to 1999. • Always use the **permit** keyword for ACLs used as match criteria in QoS policies. QoS policies do not match ACLs that use the **deny** keyword. • For *source*, enter the network or host from which the packet is being sent. You can use the **any** keyword as an abbreviation for 0.0.0.0 255.255.255.255. • (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source.
or	**ip access-list standard** *name*	Define a standard IPv4 access list using a name, and enter access-list configuration mode. The name can be a number from 1 to 99. In access-list configuration mode, enter **permit** *source* [*source-wildcard*]
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show access-lists**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To delete an access list, use the **no access-list** *access-list-number* global configuration command.

This example shows how to allow access for only those hosts on the three specified networks. The wildcard bits apply to the host portions of the network addresses.

```
Switch(config)# access-list 1 permit 192.5.255.0 0.0.0.255
Switch(config)# access-list 1 permit 128.88.0.0 0.0.255.255
Switch(config)# access-list 1 permit 36.0.0.0 0.0.0.255
```

Creating IP Extended ACLs

Beginning in privileged EXEC mode, follow these steps to create an IP extended ACL for IP traffic:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**access-list** *access-list-number* **permit** *protocol* {*source source-wildcard destination destination-wildcard*} [**precedence** *precedence*] [**tos** *tos*] [**dscp** *dscp*] **Note** If you enter a **dscp** value, you cannot enter **tos** or **precedence**. You can enter both a **tos** and a **precedence** value with no **dscp**.	Create an IP extended ACL, repeating the command as many times as necessary. • For *access-list-number*, enter the access list number. The range is 100 to 199 and 2000 to 2699. • Always use the **permit** keyword for ACLs used as match criteria in QoS policies. QoS policies do not match ACLs that use the **deny** keyword. • For *protocol*, enter the name or number of an IP protocol. Use the question mark (?) to see a list of available protocol keywords. To match any Internet protocol (including ICMP, TCP, and UDP), use the keyword **ip**. • The *source* is the number of the network or host from which the packet is sent. • The *source-wildcard* applies wildcard bits to the source. • The *destination* is the network or host number to which the packet is sent. • The *destination-wildcard* applies wildcard bits to the destination. Source, source-wildcard, destination, and destination-wildcard can be specified as: • The 32-bit quantity in dotted-decimal format. • The keyword **any** for 0.0.0.0 255.255.255.255 (any host). • The keyword **host** for a single host 0.0.0.0. Other keywords are optional and have these meanings: • **precedence**—Enter to match packets with a precedence level specified as a number from 0 to 7 or by name: **routine** (**0**), **priority** (**1**), **immediate** (**2**), **flash** (**3**), **flash-override** (**4**), **critical** (**5**), **internet** (**6**), **network** (**7**). • **tos**—Enter to match by type of service level, specified by a number from 0 to 15 or a name: **normal** (**0**), **max-reliability** (**2**), **max-throughput** (**4**), **min-delay** (**8**). • **dscp**—Enter to match packets with the DSCP value specified by a number from 0 to 63, or use the question mark (?) to see a list of available values.
or	**ip access-list extended** *name*	Define an extended IPv4 access list using a name, and enter access-list configuration mode. The *name* can be a number from 100 to 199. In access-list configuration mode, enter **permit** *protocol* {*source source-wildcard destination destination-wildcard*} [**precedence** *precedence*] [**tos** *tos*] [**dscp** *dscp*] as defined in Step 2.

	Command	Purpose
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show access-lists**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To delete an access list, use the **no access-list** *access-list-number* global configuration command.

This example shows how to create an ACL that permits IP traffic from any source to any destination that has the DSCP value set to 32:

```
Switch(config)# access-list 100 permit ip any any dscp 32
```

This example shows how to create an ACL that permits IP traffic from a source host at 10.1.1.1 to a destination host at 10.1.1.2 with a precedence value of 5:

```
Switch(config)# access-list 100 permit ip host 10.1.1.1 host 10.1.1.2 precedence 5
```

Creating Layer 2 MAC ACLs

Beginning in privileged EXEC mode, follow these steps to create a Layer 2 MAC ACL for non-IP traffic:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mac access-list extended** *name*	Create a Layer 2 MAC ACL by specifying the name of the list and enter extended MAC ACL configuration mode.
Step 3	**permit** {**host** *src-MAC-addr mask* \| **any** \| **host** *dst-MAC-addr* \| *dst-MAC-addr mask*} [*type mask*]	Always use the **permit** keyword for ACLs used as match criteria in QoS policies. • For *src-MAC-addr*, enter the MAC address of the host from which the packet is being sent. You can specify in hexadecimal format (H.H.H), use the **any** keyword for *source* 0.0.0, *source-wildcard* ffff.ffff.ffff, or use the **host** keyword for *source* 0.0.0. • For *mask*, enter the wildcard bits by placing ones in the bit positions that you want to ignore. • For *dst-MAC-addr*, enter the MAC address of the host to which the packet is being sent. You can specify in hexadecimal format (H.H.H), use the **any** keyword for *source* 0.0.0, *source-wildcard* ffff.ffff.ffff, or use the **host** keyword for *source* 0.0.0. • (Optional) For *type mask*, specify the Ethertype number of a packet with Ethernet II or SNAP encapsulation to identify the protocol of the packet. For *type*, the range is from 0 to 65535, typically specified in hexadecimal. For *mask*, enter the *don't care* bits applied to the Ethertype before testing for a match.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show access-lists** [*access-list-number* \| *access-list-name*]	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To delete an access list, use the **no mac access-list extended** *access-list-name* global configuration command.

This example shows how to create a Layer 2 MAC ACL with two **permit** statements. The first statement allows traffic from the host with MAC address 0001.0000.0001 to the host with MAC address 0002.0000.0001. The second statement allows only Ethertype XNS-IDP traffic from the host with MAC address 0001.0000.0002 to the host with MAC address 0002.0000.0002.

```
Switch(config)# mac access-list extended maclist1
Switch(config-ext-macl)# permit 0001.0000.0001 0.0.0 0002.0000.0001 0.0.0
Switch(config-ext-macl)# permit 0001.0000.0002 0.0.0 0002.0000.0002 0.0.0 xns-idp
Switch(config-ext-macl)# exit
```

Using Class Maps to Define a Traffic Class

You use the **class-map** global configuration command to name and to isolate a specific traffic flow (or class) from all other traffic. A class map defines the criteria to use to match against a specific traffic flow to further classify it. Match statements can include criteria such as an ACL, CoS value, DSCP value, IP precedence values, or QoS group values. You define match criterion with one or more **match** statements entered in the class-map configuration mode.

Follow these guidelines when configuring class maps:

- A **match-all** class map cannot have more than one classification criterion (one match statement), but a **match-any** class map can contain multiple match statements.
- A **match cos** command is supported only on Layer 2 IEEE 802.1Q trunk ports.
- For an input policy map, you cannot configure an IP classification (**match ip dscp**, **match ip precedence**, **match ip acl**) and a non-IP classification (**match cos** or **match mac acl**) in the same policy map or class map.
- You cannot configure **match qos-group** for an input policy map.
- In an output policy map, no two class maps can have the same classification criteria; that is, the same match qualifiers and values.
- The maximum number of class maps on the switch is 256.

Beginning in privileged EXEC mode, follow these steps to create a class map and to define the match criterion to classify traffic:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**class-map** [**match-all** \| **match-any**] {*class-map-name* \| **class-default**}	Create a class map, and enter class-map configuration mode. By default, no class maps are defined. • (Optional) Use the **match-all** keyword to perform a logical-AND of all matching statements under this class map. All match criteria in the class map must be matched. • (Optional) Use the **match-any** keyword to perform a logical-OR of all matching statements under this class map. One or more match criteria must be matched. • For *class-map-name*, specify the name of the class map. • Enter **class-default** to match all unclassified packets. If no matching statements are specified, the default is **match-all**. **Note** A **match-all** class map cannot have more than one classification criterion (match statement).
Step 3	**match** {**access-group** *acl-index-or-name* \| **cos** *cos-list* \| **ip dscp** *dscp-list* \| **ip precedence** *ip-precedence-list* \| **qos-group** *value*}	Define the match criterion to classify traffic. By default, no match criterion is defined. Only one match type per class map is supported, and only one ACL per class map is supported. • For **access-group** *acl-index-or-name*, specify the number or name of an ACL. Matching access groups is supported only in input policy maps. • For **cos** *cos-list*, enter a list of up to four CoS values in a single line to match against incoming packets. Separate each value with a space. You can enter multiple *cos-list* lines to match more than four CoS values. The range is 0 to 7. • For **ip dscp** *dscp-list*, enter a list of up to eight IPv4 DSCP values to match against incoming packets. Separate each value with a space. You can enter multiple *dscp-list* lines to match more than eight DSCP values. The numerical range is 0 to 63. You can also configure DSCP values in other forms. See the "Classification Based on IP DSCP" section on page 30-8. • For **ip precedence** *ip-precedence-list*, enter a list of up to four IPv4 precedence values to match against incoming packets. Separate each value with a space. You can enter multiple *ip-precedence-list* lines to match more than four precedence values. The range is 0 to 7. • For **qos-group** *valu*e, specify the QoS group number. The range is 0 to15. Matching of QoS groups is supported only in output policy maps.
Step 4	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 5	**show class-map**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no** form of the appropriate command to delete an existing class map or remove a match criterion.

This example shows how to create access list 103 and configure the class map called *class1*. The *class1* has one match criterion, which is access list 103. It permits traffic from any host to any destination that matches a DSCP value of 10.

```
Switch(config)# access-list 103 permit any any dscp 10
Switch(config)# class-map class1
Switch(config-cmap)# match access-group 103
Switch(config-cmap)# exit
```

This example shows how to create a class map called *class2*, which matches incoming traffic with DSCP values of 10, 11, and 12.

```
Switch(config)# class-map match-any class2
Switch(config-cmap)# match ip dscp 10 11 12
Switch(config-cmap)# exit
```

This example shows how to create a class map called *class3*, which matches incoming traffic with IP-precedence values of 5, 6, and 7:

```
Switch(config)# class-map match-any class3
Switch(config-cmap)# match ip precedence 5 6 7
Switch(config-cmap)# exit
```

Configuring Table Maps

You can configure table maps to manage a large number of traffic flows with a single command. You use table maps to correlate specific DSCP, IP precedence and CoS values to each other, to mark down a DSCP, IP precedence, or CoS value, or to assign default values. You can specify table maps in **set** commands and use them as mark-down mapping for the policers.

These table maps are supported on the switch:

- DSCP to CoS, precedence, or DSCP
- CoS to DSCP, precedence, or CoS
- Precedence to CoS, DSCP, or precedence

Note these guidelines when configuring table maps:

- The switch supports a maximum of 256 unique table maps.
- The maximum number of map statements within a table map is 64.
- Table maps cannot be used in output policy maps.

Beginning in privileged EXEC mode, follow these steps to create a table map:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**table-map** *table-map-name*	Create a table map by entering a table-map name and entering table-map configuration mode.
Step 3	**map from** *from-value* **to** *to-value*	Enter the mapping values to be included in the table. For example, if the table map is a DSCP-to-CoS table map, the *from-value* would be the DSCP value and the *to_value* would be the CoS value. Both ranges are from 0 to 63 Enter this command multiple times to include all the values that you want to map.
Step 4	**default** {*default-value* \| **copy** \| **ignore**}	Set the default behavior for a value not found in the table map. • Enter a *default-value* to specify a certain value. For example, in a DSCP-to-CoS table map, this would be a specific CoS value to apply to all unmapped DSCP values. The range is from 0 to 63. • Enter **copy** to map unmapped values to an equivalent value. In a DSCP-to-CoS table map, this command maps all unmapped DSCP values to the equivalent CoS value. • Enter **ignore** to leave unmapped values unchanged. In a DSCP-to-CoS table map, the switch does not change the CoS value of unmapped DSCP values.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show table-map** [*table-map-name*]	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To delete a table map, use the **no table-map** *table-map-name* global configuration command.

This example shows how to create a DSCP to CoS table map. A complete table would typically include additional map statements for the higher DSCP values. The default of 4 in this table means that unmapped DSCP values will be assigned a CoS value of 4.

```
Switch(config)# table-map dscp-to-cos
Switch(config-tablemap)# map from 1 to 1
Switch(config-tablemap)# map from 2 to 1
Switch(config-tablemap)# map from 3 to 1
Switch(config-tablemap)# map from 4 to 2
Switch(config-tablemap)# map from 5 to 2
Switch(config-tablemap)# map from 6 to 3
Switch(config-tablemap)# default 4
Switch(config-tablemap)# end
Switch# show table-map dscp-to-cos
```

Attaching a Traffic Policy to an Interface

You use the **service-policy** interface configuration command to attach a traffic policy to an interface and to specify the direction in which the policy should be applied: either an input policy map for incoming traffic or an output policy map for outgoing traffic. Input and output policy maps support different QoS features. See the "Configuring Input Policy Maps" section on page 30-35 and the "Configuring Output Policy Maps" section on page 30-43 for restrictions on input and output policy maps.

You can attach a service policy only to a physical port. You can attach only one input policy map and one output policy map per port.

Beginning in privileged EXEC mode, follow these steps to attach a policy map to a port:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the port to attach to the policy map, and enter interface configuration mode. Valid interfaces are physical ports.
Step 3	**service-policy** {**input** \| **output**} *policy-map-name*	Specify the policy-map name and whether it is an input policy map or an output policy map.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show policy-map interface** [*interface-id*]	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the policy map and port association, use the **no service-policy** {**input** | **output**} *policy-map-name* interface configuration command.

Configuring Input Policy Maps

Policy maps specify which traffic class to act on and what actions to take. All traffic that fails to meet matching criteria of a traffic class belongs to the default class. Input policy maps regulate traffic entering the switch. In an input policy, you can match CoS, DSCP, IP precedence, or ACLs, and configure individual policing, aggregate policing, or marking to a CoS, DSCP, IP precedence, or QoS group value.

Follow these guidelines when configuring input policy maps:

- You can attach only one input policy map per port.
- The maximum number of policy maps configured on the switch is 256.
- The maximum number of classes in each input policy map is 32.
- The number of input policy maps that can be attached in a switch is limited by the availability of hardware resources. If you attempt to attach an input policy map that causes any hardware resource limitation to be exceeded, the configuration fails.
- After you have attached an input policy map to an interface by using the **service-policy input** interface configuration command, you can modify the policy without detaching it from the interface. You can add or delete classification criteria, add or delete classes, add or delete actions, or change the parameters of the configured actions (policers, rates, mapping, marking, and so on).
- You cannot configure hierarchical policy maps as input policy maps.

- When an input policy map with only Layer 2 classification is attached to a routed port or a switch port containing a routed SVI, the service policy acts only on switching eligible traffic and not on routing eligible traffic.
- On an IEEE 802.1Q tunnel port, you can use only an input policy map with Layer 2 classification based on MAC ACLs to classify traffic. Input policy maps with Layer 3 classification or match Layer 2 CoS classification are not supported on tunnel ports.
- Input policy maps support policing and marking, not scheduling or queuing. You cannot configure **bandwidth**, **priority**, **queue-limit**, or **shape average** in input policy maps.

These sections describe how to configure different types of input policy maps:

- Configuring Input Policy Maps with Individual Policing, page 30-36
- Configuring Input Policy Maps with Aggregate Policing, page 30-39
- Configuring Input Policy Maps with Marking, page 30-41

Configuring Input Policy Maps with Individual Policing

You use the **police** policy-map class configuration command to configure individual policers to define the committed rate limitations, committed burst size limitations of the traffic, and the action to take for a class of traffic.

Follow these guidelines when configuring individual policers:

- Policing is supported only on input policy maps.
- The maximum number of policers per port is 48. The maximum number of policer instances on the switch is 229.
- You cannot configure individual policing for the class **class-default**.
- You can configure only one **exceed-action** police markdown table map of each type (CoS, DSCP, or IP precedence) on the switch. You can reference that table map in multiple policers.
- You cannot configure marking simultaneously for both conform-action and exceed-action for the same policer instance.
- When you use a table map for police exceed-action in an input policy map, the protocol type of the *map from* type of action must be the same as the protocol type of the associated classification. For example, if the associated class map represents an IP classification, the **map from** type of action that references the table map must be **dscp** or **precedence**. If the associated class map represents a non-IP classification, the **map from** type of action that references the table map must be **cos**.

Beginning in privileged EXEC mode, follow these steps to create an input policy map with individual policing:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**policy-map** *policy-map-name*	Create a policy map by entering the policy map name, and enter policy-map configuration mode. By default, no class maps are defined.
Step 3	**class** *class-map-name*	Enter a class-map name and enter policy-map class configuration mode. You must have already created the class map by using the **class-map** global configuration command.

	Command	Purpose
Step 4	**police** {*rate-bps* \| **cir** *cir-bps*} [*burst-bytes* \| **bc** *burst-bytes*]	Define a policer for the class of traffic. By default, no policer is defined. • For *rate-bps*, specify average traffic rate in bits per second (bps). The range is 8000 to 1000000000. • For **cir** *cir-bps*, specify a committed information rate (CIR) in bits per second (bps). The range is 8000 to 1000000000. • For *burst-bytes* (optional), specify the normal burst size in bytes. The range is 8000 to 1000000. • For **bc** *burst-bytes* (optional), specify the conformed burst (bc) or the number of acceptable burst bytes. The range is 8000 to 1000000.
Step 5	**conform-action** [**set-cos-transmit** *new-cos-value* **\| set-dscp-transmit** *new-dscp-value*\| **set-prec-transmit** *new-precedence-value* **\| set-qos-transmit** *qos-group-value* \| **transmit**]	(Optional) Enter the action to be taken on packets that conform to the CIR. The action can be to set a new CoS, DSCP, or IP precedence value, to set a QoS group value, or to send the packet. If no action is entered, the default action is to send the packet. **Note** You can enter a single conform-action as part of the command string following the police command. You can also enter a carriage return after the **police** command to enter policy-map class police configuration mode, where you can enter multiple actions. In policy-map class police configuration mode, you must enter an action to take.
Step 6	**exceed-action** [**drop \|** [**set-cos-transmit cos \| set-dscp-transmit dscp** \| **set-prec-transmit precedence** [**table** *table-map name*]]	(Optional) Enter the action to be taken for packets that do not conform to the CIR. The action can be to drop the packet or to mark down the packet with the help of a table map. If no action is entered, the default action is to drop the packet. **Note** You can enter a single exceed-action as part of the command string following the **police** command. Or you can enter a carriage return after the **police** command to enter policy-map class police configuration mode, where you can enter multiple actions. In policy-map class police configuration mode, you must enter an action to take.
Step 7	**exit**	Return to policy-map configuration mode.
Step 8	**exit**	Return to global configuration mode.
Step 9	**interface** *interface-id*	Enter interface configuration mode for the interface to which you want to attach the policy.
Step 10	**service-policy input** *policy-map-name*	Attach the policy map (created in Step 2) to the ingress interface.
Step 11	**end**	Return to privileged EXEC mode.
Step 12	**show policy-map** [*policy-map-name* [**class** *class-map-name*]]	Verify your entries.
Step 13	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

After you have created an input policy map, you attach it to an interface in the input direction. See the "Attaching a Traffic Policy to an Interface" section on page 30-35.

Use the **no** form of the appropriate command to delete an existing policy map, class map, or policer.

This example shows how to create a traffic classification with a CoS value of 4, create a policy map, and attach it to an ingress port. The average traffic rate is limited to 10000000 bps with a burst size of 10000 bytes:

```
Switch(config)# class-map video-class
Switch(config-cmap)# match cos 4
Switch(config-cmap)# exit
Switch(config)# policy-map video-policy
Switch(config-pmap)# class video-class
Switch(config-pmap-c)# police 10000000 10000
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface fastethernet0/1
Switch(config-if)# service-policy input video-policy
Switch(config-if)# exit
```

This example shows how to create policy map with a conform action of **set dscp** and a default exceed action.

```
Switch(config)# class-map in-class-1
Switch(config-cmap)# match dscp 14
Switch(config-cmap)# exit
Switch(config)# policy-map in-policy
Switch(config-pmap)# class in-class-1
Switch(config-pmap-c)# police 230000 8000 conform-action set-dscp-transmit 33
exceed-action drop
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface fastethernet0/1
Switch(config-if)# service-policy input in-policy
Switch(config-if)# exit
```

This example shows how to use policy-map class police configuration mode to set multiple conform actions and an exceed action. The policy map sets a committed information rate of 23000 bits per second (bps) and a conform burst size of 10000 bytes. The policy map includes multiple conform actions (for DSCP and for Layer 2 CoS) and an exceed action.

```
Switch(config)# class-map cos-set-1
Switch(config-cmap)# match cos 3
Switch(config-cmap)# exit
Switch(config)# policy-map map1
Switch(config-pmap)# class cos-set-1
Switch(config-pmap-c)# police cir 23000 bc 10000
Switch(config-pmap-c-police)# conform-action set-dscp-transmit 48
Switch(config-pmap-c-police)# conform-action set-cos-transmit 5
Switch(config-pmap-c-police)# exceed-action drop
Switch(config-pmap-c-police)# exit
Switch(config-pmap)# exit
Switch(config)# interface fastethernet0/1
Switch(config-if)# service-policy input map1
Switch(config-if)# exit
```

This example shows how to use policy-map class police configuration mode to set exceed action mark-down using table-maps. The policy map sets a committed information rate of 23000 bps and a conform burst-size of 10000 bytes. The policy map includes the default conform action (**transmit**) and the exceed action to mark the Layer 2 CoS value based on the table map.

```
Switch(config)# policy-map in-policy
Switch(config-pmap)# class in-class-1
Switch(config-pmap-c)# police cir 23000 bc 10000
Switch(config-pmap-c-police)# exceed-action set-cos-transmit cos table
police-cos-markdn-tablemap
Switch(config-pmap-c-police)# exit
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface fastethernet0/1
Switch(config-if)# service-policy input in-policy
Switch(config-if)# exit
```

Configuring Input Policy Maps with Aggregate Policing

You use the **policer aggregate** global configuration command to configure an aggregate policer. An aggregate policer is shared by multiple traffic classes within the same policy map. You define the aggregate policer, create a policy map, associate a class map with the policy map, associate the policy map with the aggregate policer, and apply the service policy to a port.

Follow these guidelines when configuring aggregate policers:

- Aggregate policing is supported only on input policy maps.
- The maximum number of policers per port is 48. The maximum number of policer instances on the switch is 229.
- The maximum number of configured aggregate policers is 256.
- You cannot configure aggregate policing for the class **class-default**.
- You can configure only one **exceed-action** police markdown table map of each type (CoS, DSCP, or IP precedence) on the switch. You can reference that table map in multiple policers.
- You cannot configure marking simultaneously for both conform-action and exceed-action for the same policer instance.
- Only one policy map can use any specific aggregate policer. Aggregate policing cannot be used to aggregate streams across multiple interfaces. You can use aggregate policing only to aggregate streams across multiple classes in a policy map attached to an interface.
- When you use a table map for police exceed-action in an input policy map, the protocol type of the map from type of action must be the same as the protocol type of the associated classification. For example, if the associated class map represents an IP classification, the **map from** type of action that references the table map must be either **dscp** or **precedence**. If the associated class map represents a non-IP classification, the **map from** type of action that references the table map must be **cos**.

Beginning in privileged EXEC mode, follow these steps to create an aggregate policer:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**policer aggregate** *aggregate-policer-name* {*rate-bps* \| **cir** *cir-bps*} [**bc** *burst- value*] [**conform-action** [**set-cos-transmit** *new-cos-value* \| **set-dscp-transmit** *new-dscp-value*\| **set-prec-transmit** *new-precedence-value* \| **set-qos-transmit** *qos-group-value* \| **transmit**] [**exceed-action** {**drop** \| [**set-cos-transmit** cos \| **set-dscp-transmit dscp** \| **set-prec-transmit precedence**] [**table** *table-map name*]}	Define the policer parameters that can be applied to multiple traffic classes within the same policy map. • For *aggregate-policer-name*, specify the name of the aggregate policer. • For *rate-bps*, specify average traffic rate in bits per second (bps). The range is 8000 to 1000000000. • For **cir** *cir-bps*, specify the committed information rate in bits per second. The range is 8000 to 1000000000 bps. • *(*Optional) For **bc** *burst-value*, specify conform burst and the number of acceptable burst bytes. The range is 8000 to 1000000 bytes. • (Optional) For **conform-action**, specify the action to take on packets that conform to the CIR. The default is to send the packet. • (Optional) For **exceed-action**, specify the action to take on packets that exceed the CIR. The default is to drop the packet. See the command reference for this release for definitions of the available keywords.
Step 3	**policy-map** *policy-map-name*	Create a policy map by entering the policy map name, and enter policy-map configuration mode.
Step 4	**class** *class-map-name*	Enter a class-map name and enter policy-map class configuration mode.
Step 5	**police aggregate** *aggregate-policer-name*	Apply an aggregate policer to multiple classes in the same policy map. For *aggregate-policer-name*, enter the name specified in Step 2.
Step 6	**exit**	Return to policy-map configuration mode.
Step 7	**exit**	Return to global configuration mode.
Step 8	**interface** *interface-id*	Enter interface configuration mode for the interface to which you want to attach the policy.
Step 9	**service-policy input** *policy-map-name*	Attach the policy map (created in Step 3) to the ingress interface.
Step 10	**end**	Return to privileged EXEC mode.
Step 11	**end**	Return to privileged EXEC mode.
Step 12	**show policer aggregate** [*aggregate-policer-name*]	Verify your entries.
Step 13	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

After you have created an aggregate policer, you attach it to an ingress port. See the "Attaching a Traffic Policy to an Interface" section on page 30-35.

To remove the specified aggregate policer from a policy map, use the **no police aggregate** *aggregate-policer-name* policy map configuration mode. To delete an aggregate policer and its parameters, use the **no policer aggregate** *aggregate-policer-name* global configuration command.

This example shows how to create an aggregate policer and attach it to multiple classes within a policy map. The policy map is attached to an ingress port.

```
Switch(config)# policer aggregate example 10900000 80000 exceed-action drop
Switch(config)# class-map testclass1
Switch(config-cmap)# match access-group 1
Switch(config-cmap)# exit
Switch(config)# class-map testclass2
Switch(config-cmap)# match access-group 2
Switch(config-cmap)# exit
Switch(config)# policy-map testexample
Switch(config-pmap)# class testclass
Switch(config-pmap-c)# police aggregate example
Switch(config-pmap-c)# exit
Switch(config-pmap)# class testclass2
Switch(config-pmap-c)# police aggregate example
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface fastethernet0/1
Switch(config-if)# service-policy input testexample
Switch(config-if)# exit
```

Configuring Input Policy Maps with Marking

You use the **set** policy-map class configuration command to set or modify the attributes for traffic belonging to a specific class. Follow these guidelines when configuring marking in policy maps:

- When you configure a **set qos-group** command in an input policy map, you cannot configure other **set** operations in the same class (such as **set dscp**, **set cos**, or **set precedence**).
- You can configure a maximum of 16 QoS groups on the switch.
- You can configure only one **set** action with a table map in a class. You cannot configure any other **set** action in the same class as a **set** action with table map.
- When you use a table map for marking in an input policy map, the protocol type of the map from type of action must be the same as the protocol type of the associated classification. For example, if the associated class map represents an IP classification, the **map from** type of action that references the table map must be either **dscp** or **precedence**. If the associated class map represents a non-IP classification, the **map from** type of action that references the table map must be **cos**.

Beginning in privileged EXEC mode, follow these steps to create an input policy map that marks traffic:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**policy-map** *policy-map-name*	Create a policy map by entering the policy map name, and enter policy-map configuration mode.

	Command	Purpose
Step 3	**class** {*class-map-name* \| **class-default**}	Enter a class-map name or **class-default**, and enter policy-map class configuration mode. • For *class-map-name*, specify the name of the class map. • Enter **class-default** to match all unclassified packets. If you enter a class-map name, you must have already created the class map by using the **class-map** global configuration command.
Step 4	**set cos** {*cos_value* \| **cos** [**table** *table-map-name*] \| **dscp** [**table** *table-map-name*] \| **precedence** [**table** *table-map-name*]} or **set** [**ip**] **dscp** {*dscp_value* \| **cos** [**table** *table-map-name*] \| **dscp** [**table** *table-map-name*] \| **precedence** [**table** *table-map-name*]} or **set** [**ip**] **precedence** {*precedence_value* \| **cos** [**table** *table-map-name*] \| **dscp** [**table** *table-map-name*] \| **precedence** [**table** *table-map-name*]} or **set qos-group** *value*	Mark traffic by setting a new value in the packet, specifying a table map, or specifying a QoS group. • For **cos** *cos_value*, enter a new CoS value to be assigned to the classified traffic. The range is 0 to 7. • For [**ip**] **dscp** *new-dscp*, enter a new DSCP value to be assigned to the classified traffic. The range is 0 to 63. • For [**ip**] **precedence** *new-precedence*, enter a new IP-precedence value to be assigned to the classified traffic. The range is 0 to 7. • Or you can configure a CoS, DSCP, or IP precedence table and optionally enter the table name. If you do not enter **table** *table-map name*, the table map default behavior is **copy**. See the "Configuring Table Maps" section on page 30-33. • For **qos-group** *value*, identify a QoS group to be used at egress to identify specific packets. The range is from 0 to 15.
Step 5	**exit**	Return to policy-map configuration mode.
Step 6	**exit**	Return to global configuration mode.
Step 7	**interface** *interface-id*	Enter interface configuration mode for the interface to which you want to attach the policy.
Step 8	**service-policy input** *policy-map-name*	Attach the policy map (created in Step 2) to the ingress interface.
Step 9	**end**	Return to privileged EXEC mode.
Step 10	**show policy-map** [*policy-map-name* [**class** *class-map-name*]]	Verify your entries.
Step 11	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no** form of the appropriate command to delete a policy map or table map or remove an assigned CoS, DSCP, precedence, or QoS-group value.

This example uses a policy map to remark a packet. The first marking (the **set** command) applies to the QoS default class map that matches all traffic not matched by class *AF31-AF33* and sets all traffic to an IP DSCP value of 1. The second marking sets the traffic in classes AF31 to AF33 to an IP DSCP of 3.

```
Switch(config)# policy-map Example
Switch(config-pmap)# class class-default
Switch(config-pmap-c)# set ip dscp 1
Switch(config-pmap-c)# exit
Switch(config-pmap)# class AF31-AF33
Switch(config-pmap-c)# set ip dscp 3
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface fastethernet0/1
Switch(config-if)# service-policy input Example
Switch(config-if)# exit
```

Configuring Output Policy Maps

You use output policy maps to manage congestion avoidance, queuing, and scheduling of packets leaving the switch. The switch has four egress queues, and you use output policy maps to control the queue traffic. You configure shaping, queue-limit, and bandwidth on these queues. You can use high priority (class-based priority queuing). Policing is not supported on output policy maps, except when configuring priority with police for class-based priority queuing. Output policy map classification criteria are matching a CoS, DSCP, or IP precedence value or a QoS group.

Follow these guidelines when configuring output policy maps on physical ports:

- You can configure a maximum of three unique output policy maps across all ports on the switch. This limit does not include any policy map that is configured only for port shaping (that is, it does not have a child-policy). When a hierarchical policy map is configured as an output policy map, the child policy map is counted in the three output policy-map limit.
- Output policy maps can include a maximum of four classes, including the class **class-default**.
- All output policy maps must have the same number of class-maps defined, either 1, 2, or 3.
- All output policy maps must use the same set of classes, although the actions for each class can differ for each output policy map.
- In an output policy map, you can match only packets that are already matched or marked in an input policy map.
- You cannot configure queuing or scheduling actions under the class **class-default** in an output policy map, except when using the **shape average** command to configure port shaping.
- In an output policy map, unless priority queuing is configured, the class default receives a minimum bandwidth guarantee equal to the unconfigured bandwidth on the port.
- After you have attached an output policy map to an interface by using the **service-policy** interface configuration command, you can change only the parameters of the configured actions (rates, percentages, and so on) or add or delete classification criteria of the class map while the policy map is attached to the interface. To add or delete a class or action, you must detach the policy map from all interfaces, modify it, and then reattach it to interfaces.

 Note If you anticipate that you might need three classes in a policy map, you should define three classes when you create the policy map, even if you are not ready to use all three at that time. You cannot add a class to a policy map after it has been attached to an interface.

- When at least one output policy map is attached to a active port, other active ports without output policy maps attached might incorrectly schedule and incorrectly order traffic that uses the same classes as the attached output policy maps. We recommend attaching output policy maps to all ports that are in use. We also recommend putting any unused ports in the shutdown state by entering the **shutdown** interface configuration command. For example, if you attach an output policy map that shapes DSCP 23 traffic to a port, then DSCP traffic that is sent out of any other port without a policy map attached could be incorrectly scheduled or ordered incorrectly with respect to other traffic sent out of the same port.
- We strongly recommended that you disable port speed autonegotiation when you attach an output policy map to a port to prevent the port from autonegotiating to a rate that would make the output policy map invalid. You can configure a static port speed by using the **speed** interface configuration command. If an output policy-map is configured on a port that is set for autonegotiation and the speed autonegotiates to a value that invalidates the policy, the port is put in the error-disabled state.

- You can attach only one output policy map per port.
- The maximum number of policy maps configured on the switch is 256.

These sections describe how to configure different types of output policy maps:

- Configuring Output Policy Maps with Class-Based-Weighted-Queuing, page 30-44
- Configuring Output Policy Maps with Class-Based Shaping, page 30-46
- Configuring Output Policy Maps with Port Shaping, page 30-47
- Configuring Output Policy Maps with Class-Based Priority Queuing, page 30-48
- Configuring Output Policy Maps with Weighted Tail Drop, page 30-53

Configuring Output Policy Maps with Class-Based-Weighted-Queuing

You use the **bandwidth** policy-map class configuration command to configure class-based weighted fair queuing (CBWFQ). CBWFQ sets the relative precedence of a queue by allocating a portion of the total bandwidth that is available for the port.

Follow these guidelines when configuring CBWFQ:

- When configuring bandwidth in a policy map, all rate configurations must be in the same format, either a configured rate or a percentage.
- The total rate of the minimum bandwidth guarantees for each queue of the policy cannot exceed the total speed for the interface.
- You cannot configure CBWFQ (**bandwidth**) and traffic (**shape average**) or priority queuing (**priority**) for the same class in an output policy map.
- You cannot configure bandwidth as an absolute rate or percentage of total bandwidth when strict priority (priority without police) is configured for another class map.
- You can configure bandwidth as a percentage of remaining bandwidth only when strict priority (priority without police) is configured for another class in the output policy map.

Beginning in privileged EXEC mode, follow these steps to use CBWFQ to control bandwidth allocated to a traffic class by specifying a minimum bandwidth as a bit rate or percentage:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**policy-map** *policy-map-name*	Create a policy map by entering the policy map name, and enter policy-map configuration mode.
Step 3	**class** *class-map-name*	Enter a class-map name, and enter policy-map class configuration mode.

	Command	Purpose
Step 4	**bandwidth** {*rate* \| **percent** *value* \| **remaining percent** *value*}	Set output bandwidth limits for the policy-map class. • Enter a *rate* to set bandwidth in kilobits per second. The range is from 64 to 1000000. • Enter **percent** *value* to set bandwidth as a percentage of the total bandwidth. The range is 1 to 100 percent. • Enter **remaining percent** *value* to set bandwidth as a percentage of the remaining bandwidth. The range is 1 to 100 percent. This keyword is valid only when strict priority (priority without police) is configured for another class in the output policy map. You must specify the same units in each bandwidth configuration in an output policy (absolute rates or percentages). The total guaranteed bandwidth cannot exceed the total available rate.
Step 5	**exit**	Return to policy-map configuration mode.
Step 6	**exit**	Return to global configuration mode.
Step 7	**interface** *interface-id*	Enter interface configuration mode for the interface to which you want to attach the policy.
Step 8	**service-policy output** *policy-map-name*	Attach the policy map (created in Step 2) to the egress interface.
Step 9	**end**	Return to privileged EXEC mode.
Step 10	**show policy-map** [*policy-map-name* [**class** *class-map-name*]]	Verify your entries.
Step 11	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

After you have created an output policy map, you attach it to an egress port. See the "Attaching a Traffic Policy to an Interface" section on page 30-35.

Use the **no** form of the appropriate command to delete an existing policy map, class map, or bandwidth configuration.

This example shows how to set the precedence of a queue by allocating 25 percent of the total available bandwidth to the traffic class defined by the class map:

```
Switch(config)# policy-map gold_policy
Switch(config-pmap)# class out_class-1
Switch(config-pmap-c)# bandwidth percent 25
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# service-policy output gold_policy
Switch(config-if)# exit
```

Configuring Output Policy Maps with Class-Based Shaping

You use the **shape average** policy-map class configuration command to configure traffic shaping. Class-based shaping is a control mechanism that is applied to classes of traffic leaving an interface and uses the shape average command to limit the rate of data transmission used for the committed information rate (CIR) for the class.

Follow these guidelines when configuring class-based shaping:

- Configuring a queue for traffic shaping sets the maximum bandwidth or peak information rate (PIR) of the queue. In the Cisco ME switch, configuring traffic shaping automatically also sets the minimum bandwidth guarantee or CIR of the queue to the same value as the PIR.
- You cannot configure CBWFQ (**bandwidth**) or priority queuing (**priority**) and traffic (**shape average**) for the same class in an output policy map.
- You cannot configure traffic shaping for a traffic class when strict priority (priority without police) is configured for another class within the output policy-map.

Beginning in privileged EXEC mode, follow these steps to use class-based shaping to configure the maximum permitted average rate for a class of traffic:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**policy-map** *policy-map-name*	Create a policy map by entering the policy map name, and enter policy-map configuration mode.
Step 3	**class** *class-map-name*	Enter a class-map name, and enter policy-map class configuration mode.
Step 4	**shape average** *target bps*	Specify the average class-based shaping rate. For *target bps*, specify the average bit rate in bits per second. The range is from 64000 to 1000000000.
Step 5	**exit**	Return to policy-map configuration mode.
Step 6	**exit**	Return to global configuration mode.
Step 7	**interface** *interface-id*	Enter interface configuration mode for the interface to which you want to attach the policy.
Step 8	**service-policy output** *policy-map-name*	Attach the policy map (created in Step 2) to the egress interface.
Step 9	**end**	Return to privileged EXEC mode.
Step 10	**show policy-map** [*policy-map-name* [**class** *class-map-name*]]	Verify your entries.
Step 11	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

After you have created an output policy map, you attach it to an egress port. See the "Attaching a Traffic Policy to an Interface" section on page 30-35.

Use the **no** form of the appropriate command to delete an existing policy map or class map or to delete a class-based shaping configuration.

This example shows how to configure traffic shaping for outgoing traffic on a Fast Ethernet port so that *outclass1*, *outclass2*, and *outclass3* get a maximum of 50, 20, and 10 Mbps of the available port bandwidth. The class **class-default** gets the remaining bandwidth.

```
Switch(config)# policy-map out-policy
Switch(config-pmap)# class classout1
Switch(config-pmap-c)# shape average 50000000
Switch(config-pmap-c)# exit
Switch(config-pmap)# class classout2
Switch(config-pmap-c)# shape average 20000000
Switch(config-pmap-c)# exit
Switch(config-pmap)# class classout3
Switch(config-pmap-c)# shape average 10000000
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface fastethernet0/1
Switch(config-if)# service-policy output out-policy
Switch(config-if)# exit
```

Configuring Output Policy Maps with Port Shaping

Port shaping is applied to all traffic leaving an interface. It uses a policy map with only class default when the maximum bandwidth for the port is specified by using the **shape average** command. A child policy can be attached to the class-default in a hierarchical policy map format to specify class-based actions for the queues on the shaped port.

The total of the minimum bandwidth guarantees (CIR) for each queue of the child policy cannot exceed the total port shape rate.

Beginning in privileged EXEC mode, follow these steps to use port shaping to configure the maximum permitted average rate for a class of traffic:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**policy-map** *policy-map-name*	Create a hierarchical policy map by entering the hierarchical policy map name, and enter policy-map configuration mode for the parent policy.
Step 3	**class class-default**	Enter a policy-map class configuration mode for the default class.
Step 4	**shape average** *target bps*	Specify the average class-based shaping rate. For *target bps*, specify the average bit rate in bits per second. The range is from 64000 to 1000000000.
Step 5	**service-policy** *policy-map-name*	Specify the child policy-map to be used in the hierarchical policy map if required.
Step 6	**exit**	Return to policy-map configuration mode.
Step 7	**exit**	Return to global configuration mode.
Step 8	**interface** *interface-id*	Enter interface configuration mode for the interface to which you want to attach the policy.
Step 9	**service-policy output** *policy-map-name*	Attach the parent policy map (created in Step 2) to the egress interface.
Step 10	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 11	**show policy-map** [*policy-map-name* [**class** *class-map-name*]]	Verify your entries.
Step 12	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

After you have created the hierarchical output policy map, you attach it to an egress port. See the "Attaching a Traffic Policy to an Interface" section on page 30-35.

Use the **no** form of the appropriate command to delete an existing hierarchical policy map, to delete a port shaping configuration, or to remove the policy map from the hierarchical policy map.

This example shows how to configure port shaping by configuring a hierarchical policy map that shapes a port to 90 Mbps, allocated according to the *out-policy* policy map configured in the previous example.

```
Switch(config)# policy-map out-policy-parent
Switch(config-pmap)# class class-default
Switch(config-pmap-c)# shape average 90000000
Switch(config-pmap-c)# service-policy out-policy
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface fastethernet0/1
Switch(config-if)# service-policy output out-policy-parent
Switch(config-if)# exit
```

Configuring Output Policy Maps with Class-Based Priority Queuing

You can use the **priority** policy-map class configuration command to ensure that a particular class of traffic is given preferential treatment. With strict priority queuing, the priority queue is constantly serviced; all packets in the queue are scheduled and sent until the queue is empty. Excessive use of the priority queues can possibly delay packets in other queues and create unnecessary congestion.

You can configure strict priority queuing (priority without police), or you can configure an unconditional priority policer (priority with police). Follow these guidelines when configuring priority queuing:

- You can associate the **priority** command with a single unique class for all attached output policies on the switch.
- When you configure a traffic class as a priority queue, you can configure only **police** and **queue-limit** as other queuing actions for the same class. You cannot configure **bandwidth** or **shape average** with priority queues in the same class.

Configuring Priority Without Police

Follow these guidelines when configuring strict priority queuing (priority without police):

- You cannot configure priority queuing without policing for a traffic class when class-based shaping (**shape average**) or CBWFQ (**bandwidth**) is configured for another class within the output policy-map.
- When you configure priority queuing without policing for a traffic class, you can only configure the other queues for sharing by using the **bandwidth remaining percent** policy-map class configuration command to allocate excess bandwidth. This command does not guarantee the allocated bandwidth, but does ensure the rate of distribution.

Beginning in privileged EXEC mode, follow these steps to configure a strict priority queue:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**class-map** *class-map-name*	Create classes for three egress queues. Enter match conditions classification for each class.
Step 3	**policy-map** *policy-map-name*	Create a policy map by entering the policy map name, and enter policy-map configuration mode.
Step 4	**class** *class-map-name*	Enter the name of the priority class (created by using the **class-map** global configuration command), and enter policy-map class configuration mode for the priority class.
Step 5	**priority**	Set the strict scheduling priority for this class. **Note** Only one unique class map on the switch can be associated with a **priority** command. You cannot configure priority along with any other queuing action (**bandwidth** or **shape average**).
Step 6	**exit**	Exit policy-map class configuration mode for the priority class.
Step 7	**class** *class-map-name*	Enter the name of a nonpriority class, and enter policy-map class configuration mode for that class.
Step 8	**bandwidth remaining percent** *value*	Set output bandwidth limits for the policy-map class as a percentage of the remaining bandwidth. The range is 1 to 100 percent.
Step 9	**exit**	Exit policy-map class configuration mode for the class
Step 10	**exit**	Return to global configuration mode.
Step 11	**interface** *interface-id*	Enter interface configuration mode for the interface to which you want to attach the policy.
Step 12	**service-policy output** *policy-map-name*	Attach the policy map (created in Step 3) to the egress interface.
Step 13	**end**	Return to privileged EXEC mode.
Step 14	**show policy-map**	Verify your entries.
Step 15	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

After you have created an output policy map, you attach it to an egress port. See the "Attaching a Traffic Policy to an Interface" section on page 30-35.

Use the **no** form of the appropriate command to delete an existing policy map or class map or to cancel strict priority queuing for the priority class or the bandwidth setting for the other classes.

This example shows how to configure the class *out-class1* as a strict priority queue so that all packets in that class are sent before any other class of traffic. Other traffic queues are configured so that *out-class-2* gets 50 percent of the remaining bandwidth and *out-class3* gets 20 percent of the remaining bandwidth. The class **class-default** receives the remaining 30 percent with no guarantees.

```
Switch(config)# policy-map policy1
Switch(config-pmap)# class out-class1
Switch(config-pmap-c)# priority
Switch(config-pmap-c)# exit
Switch(config-pmap)# class out-class2
Switch(config-pmap-c)# bandwidth remaining percent 50
Switch(config-pmap-c)# exit
Switch(config-pmap)# class out-class3
Switch(config-pmap-c)# bandwidth remaining percent 20
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# service-policy output policy1
Switch(config-if)# exit
```

Configuring Priority With Police

You can use the priority with police feature and configure an unconditional priority policer to limit the bandwidth used by the priority queue and allocate bandwidth or shape other queues. Follow these guidelines when configuring priority with police:

- You cannot configure a policer committed burst size for an unconditional priority policer even though the keyword is visible in the CLI help. Any configured burst size is ignored when you try to attach the output service policy.
- The allowed police rate range is 64000 to 1000000000 bps, even though the range that appears in the CLI help is 8000 to 1000000000. You cannot attach an output service policy with an out-of-range rate.
- You cannot configure priority with policing for a traffic class when **bandwidth remaining percent** is configured for another class in the same output policy map.

Beginning in privileged EXEC mode, follow these steps to configure priority with police:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**class-map** *class-map-name*	Create classes for three egress queues. Enter match conditions classification for each class.
Step 3	**policy-map** *policy-map-name*	Create a policy map by entering the policy map name, and enter policy-map configuration mode.
Step 4	**class** *class-map-name*	Enter the name of the priority class (created by using the **class-map** global configuration command), and enter policy-map class configuration mode for the priority class.
Step 5	**priority**	Configure this class as the priority class. **Note** Only one unique class map on the switch can be associated with a **priority** command.

	Command	Purpose
Step 6	**police** {*rate-bps* \| **cir** *cir-bps*}	Define a policer for the priority class of traffic. • For *rate-bps*, specify average traffic rate in bits per second (bps). The range is 64000 to 1000000000. **Note** When you use the **police** command with the **priority** command in an output policy, the police rate range and the cir range is 64000 to 1000000000 bps, even though the range that appears in the CLI help is 8000 to 1000000000. You cannot attach an output service policy with an out-of-range rate. • For **cir** *cir-bps*, specify a committed information rate (CIR) in bits per second (bps). The range is 64000 to 1000000000. **Note** Although visible in the command-line help string, the burst-size option is not supported in output policy maps. The switch does not allow you to attach an output service policy map that has a configured burst size.
Step 7	**conform-action [transmit]**	(Optional) Enter the action to be taken on packets that conform to the CIR. If no action is entered, the default action is to send the packet. **Note** You can enter a single conform-action as part of the command string following the **police** command. You can also enter a carriage return after the **police** command and enter policy-map class police configuration mode to enter the conform-action. When the *police* command is configured with priority in an output policy map, only the default conform-action of **transmit** is supported. Although visible in the command-line help string, the other police conform actions are not supported in output policy maps.
Step 8	**exceed-action** [**drop**]	(Optional) Enter the action to be taken for packets that do not conform to the CIR. If no action is entered, the default action is to drop the packet. **Note** You can enter a single exceed-action as part of the command string following the **police** command. You can also enter a carriage return after the **police** command and enter policy-map class police configuration mode to enter the exceed-action. When the *police* command is configured with priority in an output policy map, only the default exceed-action of **drop** is supported. Although visible in the command-line help string, the other police exceed actions are not supported in output policy maps.
Step 9	**exit**	Exit policy-map class configuration mode for the priority class.
Step 10	**class** *class-map-name*	Enter the name of the first nonpriority class, and enter policy-map class configuration mode for that class.

	Command	Purpose
Step 11	**bandwidth {rate \| percent** *value*}	Set output bandwidth limits for the policy-map class in kilobits per second (the range is 64 to 1000000) or percentage of the total bandwidth (the range is 1 to 100 percent).
	or	or
	shape average *target bps*	Specify the average class-based shaping rate in bits per second (the range is 64000 to 1000000000).
Step 12	**exit**	Return to policy-map configuration mode.
Step 13	**exit**	Return to global configuration mode.
Step 14	**interface** *interface-id*	Enter interface configuration mode for the interface to which you want to attach the policy.
Step 15	**service-policy output** *policy-map-name*	Attach the policy map (created in Step 3) to the egress interface.
Step 16	**end**	Return to privileged EXEC mode.
Step 17	**show policy-map**	Verify your entries.
Step 18	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

After you have created an output policy map, you attach it to an egress port. See the "Attaching a Traffic Policy to an Interface" section on page 30-35.

Use the **no** form of the appropriate command to delete an existing policy map or class map or to cancel the priority queuing or policing for the priority class or the bandwidth setting for the other classes.

This example shows how to use the **priority** with **police** commands to configure *out-class1* as the priority queue, with traffic going to the queue limited to 20000000 bps so that the priority queue never uses more than that. Traffic above that rate is dropped. The other traffic queues are configured as in the previous example.

```
Switch(config)# policy-map policy1
Switch(config-pmap)# class out-class1
Switch(config-pmap-c)# priority
Switch(config-pmap-c)# police 200000000
Switch(config-pmap-c)# exit
Switch(config-pmap)# class out-class2
Switch(config-pmap-c)# bandwidth percent 50
Switch(config-pmap-c)# exit
Switch(config-pmap)# class out-class3
Switch(config-pmap-c)# bandwidth percent 20
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# service-policy output policy1
Switch(config-if)# exit
```

Configuring Output Policy Maps with Weighted Tail Drop

Weighted tail drop (WTD) adjusts the queue size (buffer size) associated with a traffic class. You configure WTD by using the **queue-limit** policy-map class configuration command.

Follow these guidelines when configuring WTD:

- Configuring WTD with the **queue-limit** command is supported only when you first configure a scheduling action, such as **bandwidth**, **shape average**, or **priority**.
- When you use the **queue-limit** command to configure queue thresholds for a class, the WTD thresholds must be less than or equal to the queue maximum threshold. A queue size configured with no qualifier must be larger than any queue sizes configured with qualifiers.
- You cannot configure more than two unique threshold values for the WTD qualifiers (**cos**, **dscp**, **precedence**, or **qos-group**) in the **queue-limit** command. However, there is no limit to the number of qualifiers that you can map to those thresholds. You can configure a third unique threshold value to set the maximum queue, using the **queue-limit** command with no qualifiers.
- A WTD qualifier in the **queue-limit** command must be the same as at least one **match** qualifier in the associated class map.
- In an output policy map, when you configure a queue-limit for a unique class, all other output policy maps must use the same format of qualifier type and qualifier value. Only queue-limit threshold values can be different. For example, when you configure class A queue-limit thresholds for **dscp 30** and **dscp 50** in *policy-map1*, and you configure class A queue-limits in policy-map 2, you must use **dscp 30** and **dscp 50** as qualifiers. You cannot use **dscp 20** and **dscp 40**. Only the threshold values can be different.

Beginning in privileged EXEC mode, follow these steps to use WTD to adjust the queue size for a traffic class:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**policy-map** *policy-map-name*	Create a policy map by entering the policy map name, and enter policy-map configuration mode.
Step 3	**class** *class-map-name*	Enter a class-map name created by using the **class-map** global configuration command, and enter policy-map class configuration mode.
Step 4	**bandwidth** {*rate* \| **percent** *value* \| **remaining percent** *value*} or **shape average** *target bps* or **priority**	Configure a scheduling action for the traffic class. For more information, see the "Configuring Output Policy Maps with Class-Based-Weighted-Queuing" section on page 30-44, the "Configuring Output Policy Maps with Class-Based Shaping" section on page 30-46, the "Configuring Output Policy Maps with Port Shaping" section on page 30-47, or the "Configuring Output Policy Maps with Class-Based Priority Queuing" section on page 30-48.

	Command	Purpose
Step 5	**queue-limit** [**cos** *value* \| **dscp** *value* \| **precedence** *value* \| **qos-group** *value*] *number-of-packets* [**packets**]}	Specify the queue size for the traffic class. • (Optional) For **cos** *value*, specify a CoS value. The range is from 0 to 7. • (Optional) For **dscp** *value*, specify a DSCP value. The range is from 0 to 63. • (Optional) For **precedence** *value*, specify an IP precedence value. The range is from 0 to 7. • (Optional) For **qos-group** *value*, enter a QoS group value. The range is from 0 to 15. • For *number-of-packets*, set the minimum threshold for WTD. The range is from 16 to 272, in multiples of 16, where each packet it a fixed unit of 256 bytes. The value is specified in packets by default, but the **packets** keyword is optional.
Step 6	**exit**	Return to policy-map configuration mode.
Step 7	**exit**	Return to global configuration mode.
Step 8	**interface** *interface-id*	Enter interface configuration mode for the interface to which you want to attach the policy.
Step 9	**service-policy output** *policy-map-name*	Attach the policy map (created in Step 2) to the egress interface.
Step 10	**end**	Return to privileged EXEC mode.
Step 11	**show policy-map** [*policy-map-name* [**class** *class-map-name*]]	Verify your entries.
Step 12	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

After you have created an output policy map, you attach it to an egress port. See the "Configuring Output Policy Maps" section on page 30-43.

Use the **no** form of the appropriate command to delete an existing policy map or class map or to delete a WTD configuration.

This example shows a policy map with a specified bandwidth and queue size. Traffic that is not DSCP 30 or 10 is assigned a queue limit of 112 packets. Traffic with a DSCP value of 30 is assigned a queue-limit of 48 packets, and traffic with a DSCP value of 10 is assigned a queue limit of 32 packets.

```
Switch(config)# policy-map gold-policy
Switch(config-pmap)# class traffic
Switch(config-pmap-c)# bandwidth percent 50
Switch(config-pmap-c)# queue-limit 112
Switch(config-pmap-c)# queue-limit dscp 30 48
Switch(config-pmap-c)# queue-limit dscp 10 32
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# service-policy output gold-policy
Switch(config-if)# exit
```

Displaying QoS Information

To display QoS information, use one or more of the privileged EXEC commands in Table 30-2. For explanations about available keywords, see the command reference for this release.

Table 30-2 Commands for Displaying Standard QoS Information

Command	Purpose
show class-map [*class-map-name*]	Display QoS class-map information for all class maps or the specified class map.
show policer aggregate [*aggregate-policer-name*]	Display information about all aggregate policers or the specified aggregate policer.
show policy-map [*policy-map-name* \| **interface** [*interface-id*] [**input** \| **output**] [**class** *class-name*]]	Display QoS policy map information for the specified policy map name, interface, input or output policy maps, or policy-map class.
show running-config	Display the configured class maps, policy maps, table maps, and aggregate policers.
show table-map [*table-map-name*]	Display information for all configured table maps or the specified table map.

QoS Statistics

There are several ways to display QoS input and output policy-map statistics.

For input policy maps, you can use the **show policy-map interface** [*interface-id*] privileged EXEC command to display per-class per-policer conform and exceed statistics. Policer conform statistics are the number of packets that conform to the configured policer profile; policer exceed statistics are the number of packets that exceed the configured policer profile. The switch does not support per-class classification statistics, but you can determine these statistics by configuring policing at line rate for the class. In this case, no packets exceed the configured policer profile, and the policer conform statistics would equal the class classification statistics.

Another way to view input QoS statistics is in the output of the **show platform qos statistics interface** [*interface-id*] privileged EXEC command. The per-port frame statistics are sorted by the DSCP and CoS values of the incoming frames on the port. These statistics do not provide any information about the MQC input policy map configured on the interface.

For output policy maps, you can use the **show policy-map interface** [*interface-id*] command to display per-class classification statistics that show the total number of packets that match the specified class. This count includes the total number of packets that are sent and dropped for that class. You can use the same command to view the per-class tail drop statistics.

Configuration Examples for Policy Maps

This section includes configuration examples for configuring QoS policies on the Cisco ME switch, including configuration limitations and restrictions. The sections are broken into different configurations actions that a customer might do. Each section provides the exact sequence of steps that you must follow for successful configuration or modification.

These sections are included:

- QoS Configuration for Customer A, page 30-56
- QoS Configuration for Customer B, page 30-58
- Modifying Output Policies and Adding or Deleting Classification Criteria, page 30-59
- Modifying Output Policies and Changing Queuing or Scheduling Parameters, page 30-60
- Modifying Output Policies and Adding or Deleting Configured Actions, page 30-60
- Modifying Output Policies and Adding or Deleting a Class, page 30-61

QoS Configuration for Customer A

This section provides examples of the initial configuration and activation of QoS policies for a customer switch. Input and output QoS service policies are configured based on the requirements and attached to relevant ports.

In the initial configuration for Customer A, Fast Ethernet ports 1 through 24 are user network interfaces (UNIs) and are disabled by default. Gigabit Ethernet ports 1 and 2 are network node interfaces (NNIs) and are enabled by default.

This is the overall sequence for initial configuration:

- Configure classes and policies.
- Shut down all active ports.
- Attach policies to ports to be activated.
- Take the ports out of shut-down state.
- Leave unused ports shut down.

Note these restrictions for configuring output policies:

- You can define up to three classes in the output policy map.
- The defined classes must be the same as other output policy maps.
- The number of defined classes in each output policy map must be same.
- You must assign an action to each class; that is, there can be no empty class.
- Each class configuration must be based on the classification/marking done in the input policy-map.

This example configures classes for input service policies and defines three classes of service: gold, silver, and bronze. Because a **match-all** classification (the default) can have only single classification criterion, the **match-any** classification is used so that you can add classification criteria in the future.

```
Switch# config terminal
Switch(config)# class-map match-any gold-in
Switch(config-cmap)# match ip dscp af11
Switch(config-cmap)# exit
Switch(config)# class-map match-any silver-in
Switch(config-cmap)# match ip dscp af21
Switch(config-cmap)# exit
Switch(config)# class-map match-any bronze-in
Switch(config-cmap)# match ip dscp af31
Switch(config-cmap)# exit
```

This example shows how to configure an input policy map that marks the gold class and polices the silver class to 50 Mbps and the bronze class to 20 Mbps.

```
Switch(config)# policy-map input-all
Switch(config-pmap)# class gold-in
Switch(config-pmap-c)# match ip dscp af43
Switch(config-pmap-c)# exit
Switch(config-pmap)# class silver-in
Switch(config-pmap-c)# police 50000000
Switch(config-pmap)# class bronze-in
Switch(config-pmap-c)# police 20000000
Switch(config-pmap-c)# exit
```

This example configures classes for output service policies with three classes of service: gold, silver, and bronze. The gold class is configured to match the marked value in the input service policy. Because a **match-all** classification (the default) can have only single classification criterion, the **match-any** classification is used so that you can add classification criteria in the future.

```
Switch# config terminal
Switch(config)# class-map match-any gold-out
Switch(config-cmap)# match ip dscp af43
Switch(config-cmap)# exit
Switch(config)# class-map match-any silver-out
Switch(config-cmap)# match ip dscp af21
Switch(config-cmap)# exit
Switch(config)# class-map match-any bronze-out
Switch(config-cmap)# match ip dscp af31
Switch(config-cmap)# exit
```

This example configures one output service policy to be applied to both Gigabit Ethernet NNIs, providing priority with rate-limiting to the gold class, class-based shaping for the silver class, and a minimum bandwidth guarantee of 10 percent to the bronze class.

```
Switch(config)# policy-map output-g1-2
Switch(config-pmap)# class gold-out
Switch(config-pmap-c)# priority
Switch(config-pmap-c)# police 50000000
Switch(config-pmap-c)# exit
Switch(config-pmap)# class silver-out
Switch(config-pmap-c)# shape average 200000
Switch(config-pmap-c)# exit
Switch(config-pmap)# class bronze-out
Switch(config-pmap-c)# bandwidth percent 10
Switch(config-pmap-c)# exit
```

This example configures a second output service policy to be applied to Fast Ethernet UNIs 1 to 8, providing strict priority to the gold class and distributing the remaining bandwidth in the desired proportions over the remaining classes.

```
Switch(config)# policy-map output1-8
Switch(config-pmap)# class gold-out
Switch(config-pmap-c)# priority
Switch(config-pmap-c)# exit
Switch(config-pmap)# class silver-out
Switch(config-pmap-c)# bandwidth remaining percent 50
Switch(config-pmap-c)# exit
Switch(config-pmap)# class bronze-out
Switch(config-pmap-c)# bandwidth remaining percent 20
Switch(config-pmap-c)# exit
```

This example attaches the input and output service policies to the Gigabit Ethernet ports and activates them.

```
Switch(config)# interface range gigabitethernet0/1-2
Switch(config-if-range)# service-policy input input-all
Switch(config-if-range)# service-policy output output-g1-2
Switch(config-if-range)# no shutdown
Switch(config-if-range)# exit
```

This example attaches the input and output service policies to Fast Ethernet ports 1 to 8 and activates them.

```
Switch(config)# interface range fastethernet0/1 – 8
Switch(config-if-range)# service-policy input input-all
Switch(config-if-range)# service-policy output output1-8
Switch(config-if-range)# no shutdown
Switch(config-if-range)# exit
```

QoS Configuration for Customer B

This section provides examples for configuring and activating QoS policies on the switch for a new set of customers without affecting the current customers. Input and output QoS service policies are configured based on the requirements and attached to relevant ports. The example uses an existing input policy-map and configures a new output policy map for the new customers.

In the initial configuration for Customer B, Fast Ethernet ports 1 through 8 are UNIs and are active. Fast Ethernet ports 9 through 24 are UNIs and are shut down. Gigabit Ethernet ports 1 and 2 are NNIs and are enabled by default.

This is the overall sequence of initial configuration:

- Define any new required output policies.
- Attach input and output policies to ports to be activated.
- Take the ports out of the shut-down state.

Note these restrictions when configuring output policies:

- You can define up to three classes in the output policy map.
- The defined classes must be the same as other output policy maps.
- The number of defined classes in each output policy map must be same.
- You must assign an action to each class; that is, there can be no empty class.
- Each class configuration must be based on the classification/marking done in the input policy-map.

This example configures a third output service policy to be attached to Fast Ethernet UNIs 9 through 12, providing a minimum guaranteed bandwidth of 50 Mbps to the gold class, 20 Mbps to the silver class, and 10 Mbps to the bronze class:

```
Switch(config)# policy-map output9-12
Switch(config-pmap)# class gold-out
Switch(config-pmap-c)# bandwidth 50000
Switch(config-pmap-c)# exit
Switch(config-pmap)# class silver-out
Switch(config-pmap-c)# bandwidth 20000
Switch(config-pmap-c)# exit
Switch(config-pmap)# class bronze-out
Switch(config-pmap-c)# bandwidth 10000
Switch(config-pmap-c)# exit
```

This example attaches the output policy for Fast Ethernet ports 9 through 12 and activates the ports:

```
Switch# config terminal
Switch(config)# interface range fastethernet0/9-12
Switch(config-if-range)# service-policy input input-all
Switch(config-if-range)# service-policy output output9-12
Switch(config-if-range)# no shutdown
Switch(config-if-range)# exit
```

Modifying Output Policies and Adding or Deleting Classification Criteria

This section provides examples of updating an existing set of output policy maps to add or delete classification criteria. The modification might be required due to a change in the service provisioning requirements or a change in the input service policy map. You can make the change without shutting down any port.

In the initial configuration, Fast Ethernet ports 1 through 12 are UNIs and are active. Fast Ethernet ports 13 through 24 are UNIs and are shut down. Gigabit Ethernet ports 1 and 2 are NNIs and are enabled by default.

This is the overall sequence of initial configuration:

- Change the configured class map for an input service policy.
- Change the configured class map for an output service policy.

This example modifies classes for an input service policy by adding classification criteria to the silver-in class to also match dscp cs5. This is required for the output policy-map to match to dscp cs5.

```
Switch(config)# class-map match-any silver-in
Switch(config-cmap)# match ip dscp af21
Switch(config-cmap)# match ip dscp cs5
Switch(config-cmap)# exit
```

This example modifies classes for an output service policy, adding classification criteria to the silver-out class to also match dscp cs5. This adds dscp cs5 to the silver-out class on all configured and attached output service policies. The dscp cs5 flow now receives the same queuing and scheduling treatment as the silver-out class.

```
Switch# config terminal
Switch(config)# class-map match-any silver-out
Switch(config-cmap)# match ip dscp af21
Switch(config-cmap)# match ip dscp cs5
Switch(config-cmap)# exit
```

You should use the same procedure when deleting a match statement associated with a configured class.

Modifying Output Policies and Changing Queuing or Scheduling Parameters

This section provides examples of updating an existing set of output policy maps to modify the parameters of the configured queuing and scheduling actions. The modification in the output policy map might be required due to a change in the service provisioning requirements. You can make the change without shutting down any port.

In the initial configuration, Fast Ethernet ports 1 through 12 are UNIs and are active. Fast Ethernet ports 13 through 24 are UNIs and are shut down. Gigabit Ethernet ports 1 and 2 are NNIs and are enabled by default.

The requirement is to change the action parameters.

Note these restrictions when configuring output policies:

- You can define up to three classes in the output policy map.
- The defined classes must be the same as other output policy maps.
- The number of defined classes in each output policy map must be same.
- You must assign an action to each class; that is, there can be no empty class.
- Each class configuration must be based on the classification/marking done in the input policy-map.

This example modifies the third output service policy servicing Fast Ethernet UNIs 8 through 12 by providing minimum guaranteed bandwidth of 40 Mbps to the gold class (changed from 50 Mbps), 30 Mbps to the silver class (changed from 20 Mbps), and 20 Mbps to the bronze class (changed from 10 Mbps).

```
Switch(config)# policy-map output9-12
Switch(config-pmap)# class gold-out
Switch(config-pmap-c)# bandwidth 40000
Switch(config-pmap-c)# exit
Switch(config-pmap)# class silver-out
Switch(config-pmap-c)# bandwidth 30000
Switch(config-pmap-c)# exit
Switch(config-pmap)# class bronze-out
Switch(config-pmap-c)# bandwidth 20000
Switch(config-pmap-c)# exit
```

Modifying Output Policies and Adding or Deleting Configured Actions

This section provides examples of updating an existing set of output policy maps to add or delete queuing and scheduling actions. The modification in the output policy map might be required due to a change in the service provisioning requirements. You can make the change without shutting down ports that are not configured with the output policy map to be modified. But you must shut down the ports that are configured with that output policy map. Customers not using this output policy map are not affected.

In the initial configuration, Fast Ethernet ports 1 through 12 are UNIs and are active. Fast Ethernet ports 13 through 24 are UNIs and are shut down. Gigabit Ethernet ports 1 and 2 are NNIs and are enabled by default.

This is the overall sequence of configuration:

- Shut down all active ports carrying the policy to be modified.
- Detach the output policy from all ports to which it is attached.
- Make modifications to the output policy.

- Reattach the output policy to the appropriate ports.
- Take the ports out of the shutdown state.

Note these restrictions for configuring output policies:

- You can define up to three classes in the output policy map.
- The defined classes must be the same as other output policy maps.
- The number of defined classes in each output policy map must be same.
- You must assign an action to each class; that is, there can be no empty class.
- Each class configuration must be based on the classification/marking done in the input policy-map.

These steps shut down all ports carrying the output policy, in this case only the Gigabit Ethernet ports.

```
Switch(config)# interface range gigabitEthernet0/1-2
Switch(config-if-range)# shutdown
Switch(config-if-range)# exit
```

These steps detach the output policy to be modified, in this case the one configured on the Gigabit Ethernet ports:

```
Switch(config)# interface range gigabitEthernet0/1-2
Switch(config-if-range)# no service-policy output output-g1-2
Switch(config-if-range)# exit
```

These steps modify the output service policy servicing the Gigabit Ethernet NNIs. Instead of providing a minimum bandwidth guarantee of 10 percent to the bronze class, the policy is modified to provide class-based shaping to 100000 bps.

```
Switch(config)# policy-map output-g1-2
Switch(config-pmap)# class bronze-out
Switch(config-pmap-c)# no bandwidth percent 10
Switch(config-pmap-c)# shape average 100000
Switch(config-pmap-c)# exit
```

These steps reattach the output policy to the Gigabit Ethernet ports:

```
Switch(config)# interface range gigabitethernet0/1-2
Switch(config-if-range)# service-policy output output9-12
Switch(config-if-range)# exit
```

These steps activate all Gigabit Ethernet ports:

```
Switch(config)# interface range gigabitethernet0/1-2
Switch(config-if-range)# no shutdown
Switch(config-if-range)# exit
```

Modifying Output Policies and Adding or Deleting a Class

This section provides examples of updating an existing set of output policy maps to add or delete entire classes. The modification in the output policy map might be required due to a change in the service provisioning requirements or a change in the input service policy. To make this change, you must shut down all active ports on the switch. For this kind of update to any output policy map, all customers could potentially be affected. To avoid this, we recommend that you consider possible future upgrades when you configure classes in output service policies.

In the initial configuration, Fast Ethernet ports 1 through 12 are UNIs and are active. Fast Ethernet ports 13 through 24 are UNIs and are shut down. Gigabit Ethernet ports 1 and 2 are NNIs and are enabled by default.

This is the overall sequence of configuration:

- Shut down all active ports.
- Detach the output policies from all Fast Ethernet and Gigabit Ethernet ports.
- Delete the class.
- Reattach the output policies to the Fast Ethernet and Gigabit Ethernet ports.
- Take the Fast Ethernet and Gigabit Ethernet ports out of the shutdown state.

These steps shut down all active and applicable Fast Ethernet and Gigabit Ethernet ports:

```
Switch(config)# interface range gigabitethernet0/1-2, fastethernet0/1-12
Switch(config-if-range)# shutdown
Switch(config-if-range)# exit
```

These steps detach all output policies from the affected Fast Ethernet and Gigabit Ethernet ports:

```
Switch(config)# interface range fastethernet0/1-8
Switch(config-if-range)# no service-policy output output1-8
Switch(config-if-range)# exit

Switch(config)# interface range fastethernet0/9-12
Switch(config-if-range)# no service-policy output output9-12
Switch(config-if-range)# exit


Switch(config)# interface range gigabitethernet0/1-2
Switch(config-if-range)# no service-policy output output-g1-2
Switch(config-if-range)# exit
```

These steps delete a class from all output policy maps and input policy maps; the input policy can be left attached or can be detached:

```
Switch(config)# policy-map output1-8
Switch(config-pmap)# no class bronze-out
Switch(config-pmap-c)# exit
Switch(config)# policy-map output9-12
Switch(config-pmap)# no class bronze-out
Switch(config-pmap-c)# exit
Switch(config)# policy-map output-g1-2
Switch(config-pmap)# no class bronze-out
Switch(config-pmap-c)# exit
Switch(config)# policy-map input-all
Switch(config-pmap)# no class bronze-in
Switch(config-pmap-c)# exit
```

These steps reattach all policies to the Fast Ethernet and Gigabit Ethernet ports:

```
Switch(config)# interface range fastethernet0/1-8
Switch(config-if-range)# service-policy output output1-8
Switch(config-if-range)# exit

Switch(config)# interface range fastethernet0/9-12
Switch(config-if-range)# service-policy output output9-12
Switch(config-if-range)# exit

Switch(config)# interface range gigabitethernet0/1-2
Switch(config-if-range)# service-policy output output9-12
Switch(config-if-range)# exit
```

These steps activate all applicable Fast Ethernet and Gigabit Ethernet ports:

```
Switch(config)# interface range gigabitethernet0/1-2, fastethernet0/1-12
Switch(config-if-range)# no shutdown
Switch(config-if-range)# exit
```

You should use the same procedure when adding a class to an attached output service policy.



Note Problems can occur if you do not follow the previous sequence.

When a policy map is attached to an interface, all traffic that does not explicitly match the configured class maps within the policy map should go through the default queue (class **class-default**). However, in some cases, traffic that does not explicitly match the output policy-map classes could go through more than one queue. This queuing problem can occur when you do not follow the previous procedure and do not attach an output policy to all active ports.

For example, consider this case where only two ports are configured with an output policy and we want to delete a class in the output policy.

Shut down two ports:

```
Switch(config)# interface range fastethernet0/1-2
Switch(config-if-range)# shutdown
Switch(config-if-range)# exit
```

Detach the output policy from both ports:

```
Switch(config)# interface range fastEthernet0/1-2
Switch(config-if)# no service-policy output output1-2
Switch(config-if)# exit
```

Delete a class in the output policy:

```
Switch(config)# policy-map output1-2
Switch(config-pmap)# no class bronze-out
Switch(config-pmap-c)# exit
```

Attach the output policy to only one port and not to the other:

```
Switch(config)# interface FastEthernet0/1
Switch(config-if)# service-policy output output1-2
Switch(config-if)# exit
```

Enable both ports:

```
Switch(config)# interface range fastethernet0/1-2
Switch(config-if-range)# no shutdown
Switch(config-if-range)# exit
```

At this point, when traffic leaves Fast Ethernet port 2, instead of going through a single default-queue, it goes through the same number of queues as there are classes defined in the output policy-map attached to Fast Ethernet port 1. In this case, it would be three. In some cases, packets for a flow out of Fast Ethernet port 2 might be reordered if a flow splits across more than one queue. You can avoid this problem by leaving ports in a shut-down state until an output policy is attached.



Configuring EtherChannels

This chapter describes how to configure EtherChannels on Layer 2 and Layer 3 ports on the Cisco ME 3400 Ethernet Access switch. EtherChannel provides fault-tolerant high-speed links between switches, routers, and servers. You can use it to increase the bandwidth between the wiring closets and the data center, and you can deploy it anywhere in the network where bottlenecks are likely to occur. EtherChannel provides automatic recovery for the loss of a link by redistributing the load across the remaining links. If a link fails, EtherChannel redirects traffic from the failed link to the remaining links in the channel without intervention.

Note For complete syntax and usage information for the commands used in this chapter, see the command reference for this release.

This chapter consists of these sections:

- Understanding EtherChannels, page 31-1
- Configuring EtherChannels, page 31-9
- Displaying EtherChannel, PAgP, and LACP Status, page 31-21

Understanding EtherChannels

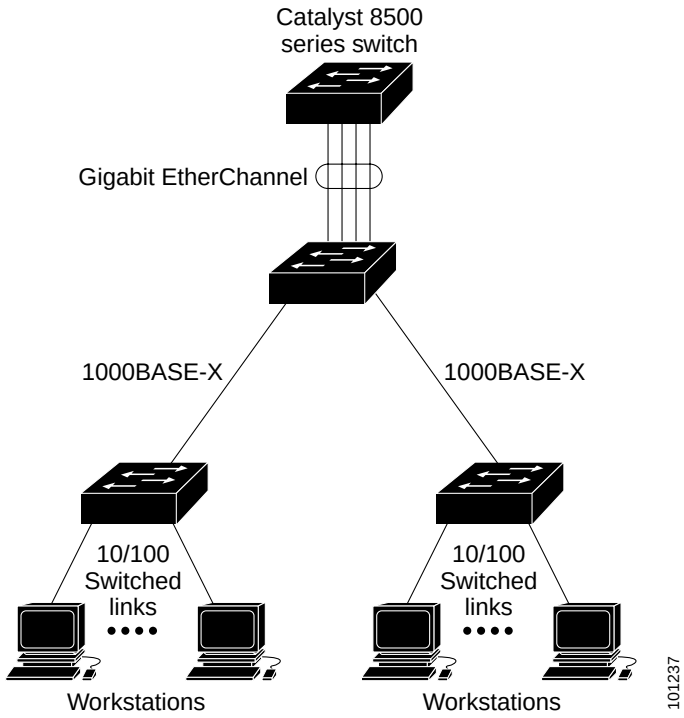
These sections describe how EtherChannels work:

- EtherChannel Overview, page 31-2
- Port-Channel Interfaces, page 31-3
- Port Aggregation Protocol, page 31-4
- Link Aggregation Control Protocol, page 31-6
- EtherChannel On Mode, page 31-7
- Load Balancing and Forwarding Methods, page 31-7

EtherChannel Overview

An EtherChannel consists of individual Fast Ethernet or Gigabit Ethernet links bundled into a single logical link as shown in Figure 31-1.

Figure 31-1 Typical EtherChannel Configuration



The EtherChannel provides full-duplex bandwidth up to 800 Mbps (Fast EtherChannel) or 2Gbps (Gigabit EtherChannel) between your switch and another switch or host.

Each EtherChannel can consist of up to eight compatibly configured Ethernet ports. All ports in each EtherChannel must be configured as either Layer 2 or Layer 3 ports. The number of EtherChannels is limited to 48. For more information, see the "EtherChannel Configuration Guidelines" section on page 31-10. The EtherChannel Layer 3 ports are made up of routed ports. Routed ports are physical ports configured to be in Layer 3 mode by using the **no switchport** interface configuration command. For more information, see the Chapter 9, "Configuring Interface Characteristics."

Note The switch must be running the metro IP access image to support Layer 3 ports.

You can configure an EtherChannel in one of these modes: Port Aggregation Protocol (PAgP), Link Aggregation Control Protocol (LACP), or On mode. PAgP and LACP are available only on NNIs. Configure both ends of the EtherChannel in the same mode:

- When you configure one end of an EtherChannel in either PAgP or LACP mode, the system negotiates with the other end of the channel to determine which ports should become active. Incompatible ports are suspended.
- When you configure an EtherChannel in the **on** mode, no negotiations take place. The switch forces all compatible ports to become active in the EtherChannel. The other end of the channel (on the other switch) must also be configured in the **on** mode; otherwise, packet loss can occur.

If a link within an EtherChannel fails, traffic previously carried over that failed link changes to the remaining links within the EtherChannel. A trap is sent for a failure, identifying the switch, the EtherChannel, and the failed link. Inbound broadcast and multicast packets on one link in an EtherChannel are blocked from returning on any other link of the EtherChannel.

Port-Channel Interfaces

When you create an EtherChannel, a port-channel logical interface is involved:

- With Layer 2 ports, use the **channel-group** interface configuration command to dynamically create the port-channel logical interface.

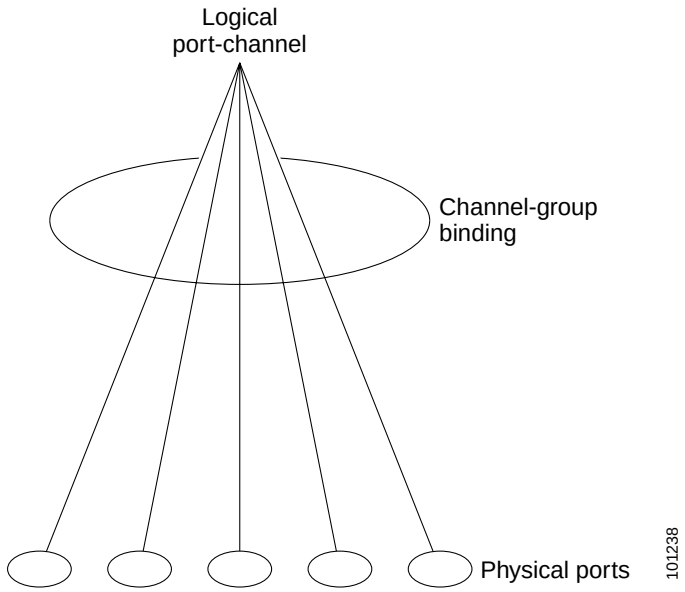
 You also can use the **interface port-channel** *port-channel-number* global configuration command to manually create the port-channel logical interface, but then you must use the **channel-group** *channel-group-number* command to bind the logical interface to a physical port. The *channel-group-number* can be the same as the *port-channel-number*, or you can use a new number. If you use a new number, the **channel-group** command dynamically creates a new port channel.
- With Layer 3 ports, you should manually create the logical interface by using the **interface port-channel** global configuration command followed by the **no switchport** interface configuration command. Then you manually assign an interface to the EtherChannel by using the **channel-group** interface configuration command.

For both Layer 2 and Layer 3 ports, the **channel-group** command binds the physical port and the logical interface together as shown in Figure 31-2.

Note The switch must be running the metro IP access image to support Layer 3 ports.

Each EtherChannel has a port-channel logical interface numbered from 1 to 48. This port-channel interface number corresponds to the one specified with the **channel-group** interface configuration command.

Figure 31-2 Relationship of Physical Ports, Logical Port Channels, and Channel Groups



After you configure an EtherChannel, configuration changes applied to the port-channel interface apply to all the physical ports assigned to the port-channel interface. Configuration changes applied to the physical port affect only the port to which you apply the configuration. To change the parameters of all ports in an EtherChannel, apply the configuration commands to the port-channel interface.

Port Aggregation Protocol

The Port Aggregation Protocol (PAgP) is a Cisco-proprietary protocol that can be run only on Cisco switches and on those switches licensed by vendors to support PAgP. PAgP facilitates the automatic creation of EtherChannels by exchanging PAgP packets between Ethernet ports.

Note PAgP is only available on network node interfaces (NNIs).

By using PAgP, the switch learns the identity of partners capable of supporting PAgP and the capabilities of each port. It then dynamically groups similarly configured ports into a single logical link (channel or aggregate port). Similarly configured ports are grouped based on hardware, administrative, and port parameter constraints. For example, PAgP groups the ports with the same speed, duplex mode, native VLAN, VLAN range, and trunking status and type. After grouping the links into an EtherChannel, PAgP adds the group to the spanning tree as a single switch port.

PAgP Modes

Table 31-1 shows the user-configurable EtherChannel PAgP modes for the **channel-group** interface configuration command on an NNI.

Table 31-1 EtherChannel PAgP Modes

Mode	Description
auto	Places a port into a passive negotiating state in which the port responds to PAgP packets it receives but does not start PAgP packet negotiation. This setting minimizes the transmission of PAgP packets.
desirable	Places a port into an active negotiating state in which the port starts negotiations with other ports by sending PAgP packets.

Switch ports exchange PAgP packets only with partner ports configured in the **auto** or **desirable** modes. Ports configured in the **on** mode do not exchange PAgP packets.

Both the **auto** and **desirable** modes enable ports to negotiate with partner ports to form an EtherChannel based on criteria such as port speed and, for Layer 2 EtherChannels, trunking state and VLAN numbers.

Ports can form an EtherChannel when they are in different PAgP modes as long as the modes are compatible. For example:

- A port in the **desirable** mode can form an EtherChannel with another port that is in the **desirable** or **auto** mode.
- A port in the **auto** mode can form an EtherChannel with another port that is in the **desirable** mode.

A port in the **auto** mode cannot form an EtherChannel with another port that is also in the **auto** mode because neither port starts PAgP negotiation.

If your switch is connected to a partner that is PAgP-capable, you can configure the switch port for nonsilent operation by using the **non-silent** keyword. If you do not specify **non-silent** with the **auto** or **desirable** mode, silent mode is assumed.

Use the silent mode when the switch is connected to a device that is not PAgP-capable and seldom, if ever, sends packets. An example of a silent partner is a file server or a packet analyzer that is not generating traffic. In this case, running PAgP on a physical port connected to a silent partner prevents that switch port from ever becoming operational. However, the silent setting allows PAgP to operate, to attach the port to a channel group, and to use the port for transmission.

PAgP Interaction with Other Features

Cisco Discovery Protocol (CDP) sends and receives packets over the physical ports in the EtherChannel. Trunk ports send and receive PAgP protocol data units (PDUs) on the lowest numbered VLAN.

Note PAgP and CDP are only available on NNIs.

In Layer 2 EtherChannels, the first port in the channel that comes up provides its MAC address to the EtherChannel. If this port is removed from the bundle, one of the remaining ports in the bundle provides its MAC address to the EtherChannel.

PAgP sends and receives PAgP PDUs only from ports that are up and have PAgP enabled for the auto or desirable mode.

Link Aggregation Control Protocol

The LACP is defined in IEEE 802.3ad standard and enables Cisco switches to manage Ethernet channels between switches that conform to the standard. LACP facilitates the automatic creation of EtherChannels by exchanging LACP packets between Ethernet ports.

Note LACP is available only on NNIs.

By using LACP, the switch learns the identity of partners capable of supporting LACP and the capabilities of each port. It then dynamically groups similarly configured ports into a single logical link (channel or aggregate port). Similarly configured ports are grouped based on hardware, administrative, and port parameter constraints. For example, LACP groups the ports with the same speed, duplex mode, native VLAN, VLAN range, and trunking status and type. After grouping the links into an EtherChannel, LACP adds the group to the spanning tree as a single switch port.

LACP Modes

Table 31-2 shows the user-configurable EtherChannel LACP modes for the **channel-group** interface configuration command on an NNI.

Table 31-2 EtherChannel LACP Modes

Mode	Description
active	Places a port into an active negotiating state in which the port starts negotiations with other ports by sending LACP packets.
passive	Places a port into a passive negotiating state in which the port responds to LACP packets that it receives, but does not start LACP packet negotiation. This setting minimizes the transmission of LACP packets.

Both the **active** and **passive** LACP modes enable ports to negotiate with partner ports to an EtherChannel based on criteria such as port speed and, for Layer 2 EtherChannels, trunking state and VLAN numbers.

Ports can form an EtherChannel when they are in different LACP modes as long as the modes are compatible. For example:

- A port in the **active** mode can form an EtherChannel with another port that is in the **active** or **passive** mode.
- A port in the **passive** mode cannot form an EtherChannel with another port that is also in the **passive** mode because neither port starts LACP negotiation.

LACP Interaction with Other Features

The CDP sends and receives packets over the physical ports in the EtherChannel. Trunk ports send and receive LACP PDUs on the lowest numbered VLAN.

In Layer 2 EtherChannels, the first port in the channel that comes up provides its MAC address to the EtherChannel. If this port is removed from the bundle, one of the remaining ports in the bundle provides its MAC address to the EtherChannel.

LACP sends and receives LACP PDUs only from ports that are up and have LACP enabled for the active or passive mode.

EtherChannel On Mode

EtherChannel **on** mode can be used to manually configure an EtherChannel. The **on** mode forces a port to join an EtherChannel without negotiations. It can be useful if the remote device does not support PAgP or LACP. With the **on** mode, a usable EtherChannel exists only when both ends of the link are configured in the **on** mode.

Note For UNIs, the only available mode is **on**.

Ports that are configured in the **on** mode in the same channel group must have compatible port characteristics, such as speed and duplex. Ports that are not compatible are suspended, even though they are configured in the **on** mode.

Caution You should use care when using the **on** mode. This is a manual configuration, and ports on both ends of the EtherChannel must have the same configuration. If the group is misconfigured, packet loss or spanning-tree loops can occur.

Load Balancing and Forwarding Methods

EtherChannel balances the traffic load across the links in a channel by reducing part of the binary pattern formed from the addresses in the frame to a numerical value that selects one of the links in the channel. EtherChannel load balancing can use MAC addresses or IP addresses, source or destination addresses, or both source and destination addresses. The selected mode applies to all EtherChannels configured on the switch. You configure the load balancing and forwarding method by using the **port-channel load-balance** global configuration command.

With source-MAC address forwarding, when packets are forwarded to an EtherChannel, they are distributed across the ports in the channel based on the source-MAC address of the incoming packet. Therefore, to provide load balancing, packets from different hosts use different ports in the channel, but packets from the same host use the same port in the channel.

With destination-MAC address forwarding, when packets are forwarded to an EtherChannel, they are distributed across the ports in the channel based on the destination-host MAC address of the incoming packet. Therefore, packets to the same destination are forwarded over the same port, and packets to a different destination are sent on a different port in the channel.

With source-and-destination MAC address forwarding, when packets are forwarded to an EtherChannel, they are distributed across the ports in the channel based on both the source and destination MAC addresses. This forwarding method, a combination source-MAC and destination-MAC address forwarding methods of load distribution, can be used if it is not clear whether source-MAC or destination-MAC address forwarding is better suited on a particular switch. With source-and-destination MAC-address forwarding, packets sent from host A to host B, host A to host C, and host C to host B could all use different ports in the channel.

With source-IP-address-based forwarding, when packets are forwarded to an EtherChannel, they are distributed across the ports in the EtherChannel based on the source-IP address of the incoming packet. Therefore, to provide load-balancing, packets from different IP addresses use different ports in the channel, but packets from the same IP address use the same port in the channel.

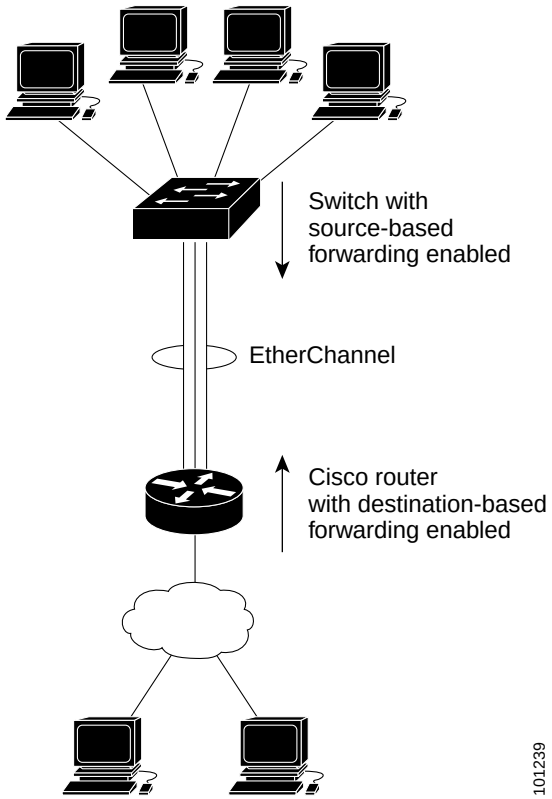
With destination-IP-address-based forwarding, when packets are forwarded to an EtherChannel, they are distributed across the ports in the EtherChannel based on the destination-IP address of the incoming packet. Therefore, to provide load-balancing, packets from the same IP source address sent to different IP destination addresses could be sent on different ports in the channel. But packets sent from different source IP addresses to the same destination IP address are always sent on the same port in the channel.

With source-and-destination IP address-based forwarding, when packets are forwarded to an EtherChannel, they are distributed across the ports in the EtherChannel based on both the source and destination IP addresses of the incoming packet. This forwarding method, a combination of source-IP and destination-IP address-based forwarding, can be used if it is not clear whether source-IP or destination-IP address-based forwarding is better suited on a particular switch. In this method, packets sent from the IP address A to IP address B, from IP address A to IP address C, and from IP address C to IP address B could all use different ports in the channel.

Different load-balancing methods have different advantages, and the choice of a particular load-balancing method should be based on the position of the switch in the network and the kind of traffic that needs to be load-distributed. In Figure 31-3, an EtherChannel of four workstations communicates with a router. Because the router is a single-MAC-address device, source-based forwarding on the switch EtherChannel ensures that the switch uses all available bandwidth to the router. The router is configured for destination-based forwarding because the large number of workstations ensures that the traffic is evenly distributed from the router EtherChannel.

Use the option that provides the greatest variety in your configuration. For example, if the traffic on a channel is going only to a single MAC address, using the destination-MAC address always chooses the same link in the channel. Using source addresses or IP addresses might result in better load balancing.

Figure 31-3 Load Distribution and Forwarding Methods



Configuring EtherChannels

These sections contain this configuration information:

- Default EtherChannel Configuration, page 31-9
- EtherChannel Configuration Guidelines, page 31-10
- Configuring Layer 2 EtherChannels, page 31-11 (required)
- Configuring Layer 3 EtherChannels, page 31-13 (required)
- Configuring EtherChannel Load Balancing, page 31-16 (optional)
- Configuring the PAgP Learn Method and Priority, page 31-17 (optional)
- Configuring LACP Hot-Standby Ports, page 31-18 (optional)

Note Make sure that the ports are correctly configured. For more information, see the "EtherChannel Configuration Guidelines" section on page 31-10.

Note After you configure an EtherChannel, configuration changes applied to the port-channel interface apply to all the physical ports assigned to the port-channel interface, and configuration changes applied to the physical port affect only the port to which you apply the configuration.

Default EtherChannel Configuration

Table 31-3 shows the default EtherChannel configuration.

Table 31-3 Default EtherChannel Configuration

Feature	Default Setting
Channel groups	None assigned.
Port-channel logical interface	None defined.
PAgP mode	No default.
PAgP learn method	Aggregate-port learning on all NNIs.
PAgP priority	128 on all NNIs.
LACP mode	No default.
LACP learn method	Aggregate-port learning on all NNIs.
LACP port priority	32768 on all NNIs.
LACP system priority	32768.
LACP system ID	LACP system priority and the switch MAC address.
Load balancing	Load distribution on the switch is based on the source-MAC address of the incoming packet.

EtherChannel Configuration Guidelines

If improperly configured, some EtherChannel ports are automatically disabled to avoid network loops and other problems. Follow these guidelines to avoid configuration problems:

- Do not try to configure more than 48 EtherChannels on the switch.
- Configure a PAgP EtherChannel including only NNIs.
- Configure a LACP EtherChannel including only NNIs.
- Configure all ports in an EtherChannel to operate at the same speeds and duplex modes.
- All ports in an EtherChannel must be the same type, either UNI or NNI. You cannot add UNIs to an EtherChannel containing active NNIs.
- On UNIs, the EtherChannel mode must always be configured to **on**.
- Enable all ports in an EtherChannel. A port in an EtherChannel that is disabled by using the **shutdown** interface configuration command is treated as a link failure, and its traffic is transferred to one of the remaining ports in the EtherChannel. UNIs are disabled by default. NNIs are enabled by default.
- When a group is first created, all ports follow the parameters set for the first port to be added to the group. If you change the configuration of one of these parameters, you must also make the changes to all ports in the group:
 - Allowed-VLAN list

 Note Spanning Tree Protocol is only supported on NNIs.

 - Spanning-tree path cost for each VLAN
 - Spanning-tree port priority for each VLAN
 - Spanning-tree Port Fast setting
- Do not configure a port to be a member of more than one EtherChannel group.
- Do not configure an EtherChannel in both the PAgP and LACP modes. EtherChannel groups running PAgP and LACP can coexist on the same switch. Individual EtherChannel groups can run either PAgP or LACP, but they cannot interoperate.
- You can only have four NNIs on the switch at the same time; therefore, only four ports in an EtherChannel can support LACP and PAgP at the same time.

Note PAgP and LACP are only available on NNIs.

- Do not configure a Switched Port Analyzer (SPAN) destination port as part of an EtherChannel.
- Do not configure a secure port as part of an EtherChannel or the reverse.
- Do not configure a private-VLAN port as part of an EtherChannel.
- Do not configure a port that is an active or a not-yet-active member of an EtherChannel as an 802.1x port. If you try to enable 802.1x on an EtherChannel port, an error message appears, and 802.1x is not enabled.
- If EtherChannels are configured on switch interfaces, remove the EtherChannel configuration from the interfaces before globally enabling 802.1x on a switch by using the **dot1x system-auth-control** global configuration command.

- For Layer 2 EtherChannels:
 - Assign all ports in the EtherChannel to the same VLAN, or configure them as trunks. Ports with different native VLANs cannot form an EtherChannel.
 - If you configure an EtherChannel from trunk ports, verify that the trunking mode is the same on all the trunks. Inconsistent trunk modes on EtherChannel ports can have unexpected results.
 - An EtherChannel supports the same allowed range of VLANs on all the ports in a trunking Layer 2 EtherChannel. If the allowed range of VLANs is not the same, the ports do not form an EtherChannel even when PAgP is set to the **auto** or **desirable** mode.
 - NNIs with different spanning-tree path costs can form an EtherChannel if they are otherwise compatibly configured. Setting different spanning-tree path costs does not, by itself, make ports incompatible for the formation of an EtherChannel.
- For Layer 3 EtherChannels, assign the Layer 3 address to the port-channel logical interface, not to the physical ports in the channel.

Configuring Layer 2 EtherChannels

You configure Layer 2 EtherChannels by assigning ports to a channel group with the **channel-group** interface configuration command. This command automatically creates the port-channel logical interface.

Beginning in privileged EXEC mode, follow these steps to assign a Layer 2 Ethernet port to a Layer 2 EtherChannel. This procedure is required.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify a physical port, and enter interface configuration mode. Valid interfaces include physical ports. For a PAgP EtherChannel, you can configure up to eight ports of the same type and speed for the same group. For a LACP EtherChannel, you can configure up to 16 Ethernet ports of the same type. Up to eight ports can be active, and up to eight ports can be in standby mode. **Note** If the interface is a UNI, you must enter the **port-type nni** interface configuration command before configuring PAgP or LACP.
Step 3	**switchport mode** {**access** \| **trunk**} **switchport access vlan** *vlan-id*	Assign all ports as static-access ports in the same VLAN, or configure them as trunks. If you configure the port as a static-access port, assign it to only one VLAN. The range is 1 to 4094.

	Command	Purpose
Step 4	**channel-group** *channel-group-number* **mode** {**auto** [**non-silent**] \| **desirable** [**non-silent**] \| **on**} \| {**active** \| **passive**}	Assign the port to a channel group, and specify the PAgP or the LACP mode. For *channel-group-number*, the range is 1 to 48. **Note** For UNIs, the only available mode is **on**. For **mode**, select one of these keywords: • **auto**—Enables PAgP only if a PAgP device is detected. It places the port into a passive negotiating state, in which the port responds to PAgP packets it receives but does not start PAgP packet negotiation. • **desirable**—Unconditionally enables PAgP. It places the port into an active negotiating state, in which the port starts negotiations with other ports by sending PAgP packets. • **on**—Forces the port to channel without PAgP or LACP. With the **on** mode, a usable EtherChannel exists only when a port group in the **on** mode is connected to another port group in the **on** mode. • **non-silent**—(Optional) If your switch is connected to a partner that is PAgP-capable, configure the switch port for nonsilent operation when the port is in the **auto** or **desirable** mode. If you do not specify **non-silent**, silent is assumed. The silent setting is for connections to file servers or packet analyzers. This setting allows PAgP to operate, to attach the port to a channel group, and to use the port for transmission. • **active**—Enables LACP only if a LACP device is detected. It places the port into an active negotiating state in which the port starts negotiations with other ports by sending LACP packets. • **passive**—Enables LACP on the port and places it into a passive negotiating state in which the port responds to LACP packets that it receives, but does not start LACP packet negotiation. For information on compatible modes for the switch and its partner, see the "PAgP Modes" section on page 31-5 and the "LACP Modes" section on page 31-6.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove a port from the EtherChannel group, use the **no channel-group** interface configuration command.

This example shows how to configure an EtherChannel. It assigns two ports as static-access ports in VLAN 10 to channel 5 with the PAgP mode **desirable**:

```
Switch# configure terminal
Switch(config)# interface range fastethernet0/1 -2
Switch(config-if-range)# port-type nni
Switch(config-if-range)# switchport mode access
Switch(config-if-range)# switchport access vlan 10
Switch(config-if-range)# channel-group 5 mode desirable non-silent
Switch(config-if-range)# end
```

This example shows how to configure an EtherChannel. It assigns two ports as static-access ports in VLAN 10 to channel 5 with the LACP mode **active**:

```
Switch# configure terminal
Switch(config)# interface range gigabitethernet0/1 -2
Switch(config-if-range)# switchport mode access
Switch(config-if-range)# switchport access vlan 10
Switch(config-if-range)# channel-group 5 mode active
Switch(config-if-range)# end
```

Configuring Layer 3 EtherChannels

To configure Layer 3 EtherChannels, you create the port-channel logical interface and then put the Ethernet ports into the port-channel as described in the next two sections.

Note The switch must be running the metro IP access image to support Layer 3 ports.

Creating Port-Channel Logical Interfaces

When configuring Layer 3 EtherChannels, you should first manually create the port-channel logical interface by using the **interface port-channel** global configuration command. Then you put the logical interface into the channel group by using the **channel-group** interface configuration command.

Note To move an IP address from a physical port to an EtherChannel, you must delete the IP address from the physical port before configuring it on the port-channel interface.

Beginning in privileged EXEC mode, follow these steps to create a port-channel interface for a Layer 3 EtherChannel. This procedure is required.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface port-channel** *port-channel-number*	Specify the port-channel logical interface, and enter interface configuration mode. For *port-channel-number*, the range is 1 to 48.
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled, and NNIs are enabled.
Step 4	**no switchport**	Put the interface into Layer 3 mode.
Step 5	**ip address** *ip-address mask*	Assign an IP address and subnet mask to the EtherChannel.

	Command	Purpose
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show etherchannel** *channel-group-number* **detail**	Verify your entries.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.
Step 9		Assign an Ethernet port to the Layer 3 EtherChannel. For more information, see the "Configuring the Physical Interfaces" section on page 31-14.

To remove the port-channel, use the **no interface port-channel** *port-channel-number* global configuration command.

This example shows how to create the logical port channel 5 and assign 172.10.20.10 as its IP address:

```
Switch# configure terminal
Switch(config)# interface port-channel 5
Switch(config-if)# no switchport
Switch(config-if)# ip address 172.10.20.10 255.255.255.0
Switch(config-if)# end
```

Configuring the Physical Interfaces

Beginning in privileged EXEC mode, follow these steps to assign an Ethernet port to a Layer 3 EtherChannel. This procedure is required.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify a physical port, and enter interface configuration mode. Valid interfaces include physical ports. For a PAgP EtherChannel, you can configure up to eight ports of the same type and speed for the same group. For a LACP EtherChannel, you can configure up to 16 Ethernet ports of the same type. Up to eight ports can be active, and up to eight ports can be in standby mode. **Note** If the interface is a UNI, you must enter the **port-type nni** interface configuration command before configuring PAgP or LACP.
Step 3	**no ip address**	Ensure that there is no IP address assigned to the physical port.
Step 4	**no switchport**	Put the port into Layer 3 mode.

	Command	Purpose
Step 5	**channel-group** *channel-group-number* **mode** {**auto** [**non-silent**] \| **desirable** [**non-silent**] \| **on**} \| {**active** \| **passive**}	Assign the port to a channel group, and specify the PAgP or the LACP mode. For *channel-group-number*, the range is 1 to 48. This number must be the same as the *port-channel-number* (logical port) configured in the "Creating Port-Channel Logical Interfaces" section on page 31-13. **Note** For UNIs, the only available mode is **on**. For **mode**, select one of these keywords: • **auto**—Enables PAgP only if a PAgP device is detected. It places the port into a passive negotiating state, in which the port responds to PAgP packets it receives but does not start PAgP packet negotiation. • **desirable**—Unconditionally enables PAgP. It places the port into an active negotiating state, in which the port starts negotiations with other ports by sending PAgP packets. • **on**—Forces the port to channel without PAgP or LACP. With the **on** mode, a usable EtherChannel exists only when a port group in the **on** mode is connected to another port group in the **on** mode. • **non-silent**—(Optional) If your switch is connected to a partner that is PAgP capable, configure the switch port for nonsilent operation when the port is in the **auto** or **desirable** mode. If you do not specify **non-silent**, silent is assumed. The silent setting is for connections to file servers or packet analyzers. This setting allows PAgP to operate, to attach the port to a channel group, and to use the port for transmission. • **active**—Enables LACP only if a LACP device is detected. It places the port into an active negotiating state in which the port starts negotiations with other ports by sending LACP packets. • **passive**—Enables LACP on the port and places it into a passive negotiating state in which the port responds to LACP packets that it receives, but does not start LACP packet negotiation. For information on compatible modes for the switch and its partner, see the "PAgP Modes" section on page 31-5 and the "LACP Modes" section on page 31-6.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show running-config**	Verify your entries.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

This example shows how to configure an EtherChannel. It assigns two ports to channel 5 with the LACP mode **active**:

```
Switch# configure terminal
Switch(config)# interface range gigabitethernet0/1 -2
Switch(config-if-range)# no ip address
Switch(config-if-range)# no switchport
Switch(config-if-range)# channel-group 5 mode active
Switch(config-if-range)# end
```

Configuring EtherChannel Load Balancing

This section describes how to configure EtherChannel load balancing by using source-based or destination-based forwarding methods. For more information, see the "Load Balancing and Forwarding Methods" section on page 31-7.

Beginning in privileged EXEC mode, follow these steps to configure EtherChannel load balancing. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**port-channel load-balance** {**dst-ip** \| **dst-mac** \| **src-dst-ip** \| **src-dst-mac** \| **src-ip** \| **src-mac**}	Configure an EtherChannel load-balancing method. The default is **src-mac**. Select one of these load-distribution methods: • **dst-ip**—Load distribution is based on the destination-host IP address. • **dst-mac**—Load distribution is based on the destination-host MAC address of the incoming packet. • **src-dst-ip**—Load distribution is based on the source-and-destination host-IP address. • **src-dst-mac**—Load distribution is based on the source-and-destination host-MAC address. • **src-ip**—Load distribution is based on the source-host IP address. • **src-mac**—Load distribution is based on the source-MAC address of the incoming packet.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show etherchannel load-balance**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return EtherChannel load balancing to the default configuration, use the **no port-channel load-balance** global configuration command.

Configuring the PAgP Learn Method and Priority

Network devices are classified as PAgP physical learners or aggregate-port learners. A device is a physical learner if it learns addresses by physical ports and directs transmissions based on that knowledge. A device is an aggregate-port learner if it learns addresses by aggregate (logical) ports. The learn method must be configured the same at both ends of the link.

Note PAgP is available only on NNIs.

When a device and its partner are both aggregate-port learners, they learn the address on the logical port-channel. The device sends packets to the source by using any of the ports in the EtherChannel. With aggregate-port learning, it is not important on which physical port the packet arrives.

PAgP cannot automatically detect when the partner device is a physical learner and when the local device is an aggregate-port learner. Therefore, you must manually set the learning method on the local device to learn addresses by physical ports. You also must set the load-distribution method to source-based distribution, so that any given source MAC address is always sent on the same physical port.

You also can configure a single port within the group for all transmissions and use other ports for hot standby. The unused ports in the group can be swapped into operation in just a few seconds if the selected single port loses hardware-signal detection. You can configure which port is always selected for packet transmission by changing its priority with the **pagp port-priority** interface configuration command. The higher the priority, the more likely that the port will be selected.

Note The switch supports address learning only on aggregate ports even though the **physical-port** keyword is provided in the CLI. The **pagp learn-method** command and the **pagp port-priority** command have no effect on the switch hardware, but they are required for PAgP interoperability with devices that only support address learning by physical ports.

When the link partner to the switch is a physical learner, we recommend that you configure the Cisco ME switch as a physical-port learner by using the **pagp learn-method physical-port** interface configuration command. Set the load-distribution method based on the source MAC address by using the **port-channel load-balance src-mac** global configuration command. The switch then sends packets to the physical learner switch using the same port in the EtherChannel from which it learned the source address. Use the **pagp learn-method** command only in this situation.

Beginning in privileged EXEC mode, follow these steps to configure your switch as a PAgP physical-port learner and to adjust the priority so that the same port in the bundle is selected for sending packets. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the port for transmission, and enter interface configuration mode.
Step 3	**pagp learn-method physical-port**	Select the PAgP learning method. By default, **aggregation-port** learning is selected, which means the switch sends packets to the source by using any of the ports in the EtherChannel. With aggregate-port learning, it is not important on which physical port the packet arrives. Select **physical-port** to connect with another switch that is a physical learner. Make sure to configure the **port-channel load-balance** global configuration command to **src-mac** as described in the "Configuring EtherChannel Load Balancing" section on page 31-16. The learning method must be configured the same at both ends of the link. **Note** If the interface is a UNI, you must enter the **port-type nni** interface configuration command before configuring PAgP.
Step 4	**pagp port-priority** *priority*	Assign a priority so that the selected port is chosen for packet transmission. For *priority*, the range is 0 to 255. The default is 128. The higher the priority, the more likely that the port will be used for PAgP transmission.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config** or **show pagp** *channel-group-number* **internal**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the priority to its default setting, use the **no pagp port-priority** interface configuration command. To return the learning method to its default setting, use the **no pagp learn-method** interface configuration command.

Configuring LACP Hot-Standby Ports

When enabled, LACP tries to configure the maximum number of LACP-compatible ports in a channel, up to a maximum of 16 ports. Only eight LACP links can be active at one time. The software places any additional links in a hot-standby mode. If one of the active links becomes inactive, a link that is in the hot-standby mode becomes active in its place.

Note LACP is only available on NNIs.

If you configure more than eight links for an EtherChannel group, the software automatically decides which of the hot-standby ports to make active based on the LACP priority. The software assigns to every link between systems that operate LACP a unique priority made up of these elements (in priority order):

- LACP system priority
- System ID (a combination of the LACP system priority and the switch MAC address)
- LACP port priority
- Port number

In priority comparisons, numerically lower values have higher priority. The priority decides which ports should be put in standby mode when there is a hardware limitation that prevents all compatible ports from aggregating.

Ports are considered for active use in aggregation in link-priority order starting with the port attached to the highest priority link. Each port is selected for active use if the preceding higher priority selections can also be maintained. Otherwise, the port is selected for standby mode.

You can change the default values of the LACP system priority and the LACP port priority to affect how the software selects active and standby links. For more information, see the "Configuring the LACP System Priority" section on page 31-19 and the "Configuring the LACP Port Priority" section on page 31-20.

Configuring the LACP System Priority

You can configure the system priority for all of the EtherChannels that are enabled for LACP by using the **lacp system-priority** global configuration command. You cannot configure a system priority for each LACP-configured channel. By changing this value from the default, you can affect how the software selects active and standby links.

You can use the **show etherchannel summary** privileged EXEC command to see which ports are in the hot-standby mode (denoted with an *H* port-state flag).

Beginning in privileged EXEC mode, follow these steps to configure the LACP system priority. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**lacp system-priority** *priority*	Configure the LACP system priority. For *priority*, the range is 1 to 65535. The default is 32768. The lower the value, the higher the system priority.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config** or **show lacp sys-id**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the LACP system priority to the default value, use the **no lacp system-priority** global configuration command.

Configuring the LACP Port Priority

By default, all ports use the same port priority. If the local system has a lower value for the system priority and the system ID than the remote system, you can affect which of the hot-standby links become active first by changing the port priority of LACP EtherChannel ports to a lower value than the default. The hot-standby ports that have lower port numbers become active in the channel first. You can use the **show etherchannel summary** privileged EXEC command to see which ports are in the hot-standby mode (denoted with an *H* port-state flag).

Note If LACP is not able to aggregate all the ports that are compatible (for example, the remote system might have more restrictive hardware limitations), all the ports that cannot be actively included in the EtherChannel are put in the hot-standby state and are used only if one of the channeled ports fails.

Beginning in privileged EXEC mode, follow these steps to configure the LACP port priority. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the port to be configured, and enter interface configuration mode. **Note** If the interface is a UNI, you must enter the **port-type nni** interface configuration command before configuring LACP.
Step 3	**lacp port-priority** *priority*	Configure the LACP port priority. For *priority*, the range is 1 to 65535. The is 32768. The lower the value, the more likely that the port will be used for LACP transmission.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config** or **show lacp** [*channel-group-number*] **internal**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the LACP port priority to the default value, use the **no lacp port-priority** interface configuration command.

Displaying EtherChannel, PAgP, and LACP Status

To display EtherChannel, PAgP, and LACP status information, use the privileged EXEC commands described in Table 31-4:

Table 31-4 Commands for Displaying EtherChannel, PAgP, and LACP Status

Command	Description
show etherchannel [*channel-group-number* {**detail** \| **port** \| **port-channel** \| **protocol** \| **summary**}] {**detail** \| **load-balance** \| **port** \| **port-channel** \| **protocol** \| **summary**}	Displays EtherChannel information in a brief, detailed, and one-line summary form. Also displays the load-balance or frame-distribution scheme, port, port-channel, and protocol information.
show pagp [*channel-group-number*] {**counters** \| **internal** \| **neighbor**}	Displays PAgP information such as traffic information, the internal PAgP configuration, and neighbor information.
show lacp [*channel-group-number*] {**counters** \| **internal** \| **neighbor**}	Displays LACP information such as traffic information, the internal LACP configuration, and neighbor information.

You can clear PAgP channel-group information and traffic counters by using the **clear pagp** {*channel-group-number* **counters** | **counters**} privileged EXEC command.

You can clear LACP channel-group information and traffic counters by using the **clear lacp** {*channel-group-number* **counters** | **counters**} privileged EXEC command.

For detailed information about the fields in the displays, see the command reference for this release.



CHAPTER 32

Configuring IP Unicast Routing

This chapter describes how to configure IP Version 4 (IPv4) unicast routing on the Cisco ME 3400 Ethernet Access switch.

Note Routing is supported only on switches that are running the metro IP access image.

For more detailed IP unicast configuration information, see the *Cisco IOS IP Configuration Guide, Release 12.2* For complete syntax and usage information for the commands used in this chapter, see these command references:

- *Cisco IOS IP Command Reference, Volume 1 of 3: Addressing and Services, Release 12.2*
- *Cisco IOS IP Command Reference, Volume 2 of 3: Routing Protocols, Release 12.2*
- *Cisco IOS IP Command Reference, Volume 3 of 3: Multicast, Release 12.2*

This chapter consists of these sections:

- Understanding IP Routing, page 32-2
- Steps for Configuring Routing, page 32-3
- Configuring IP Addressing, page 32-4
- Enabling IPv4 Unicast Routing, page 32-17
- Configuring RIP, page 32-18
- Configuring OSPF, page 32-23
- Configuring EIGRP, page 32-32
- Configuring BGP, page 32-38
- Configuring Multi-VRF CE, page 32-59
- Configuring Protocol-Independent Features, page 32-69
- Monitoring and Maintaining the IP Network, page 32-83

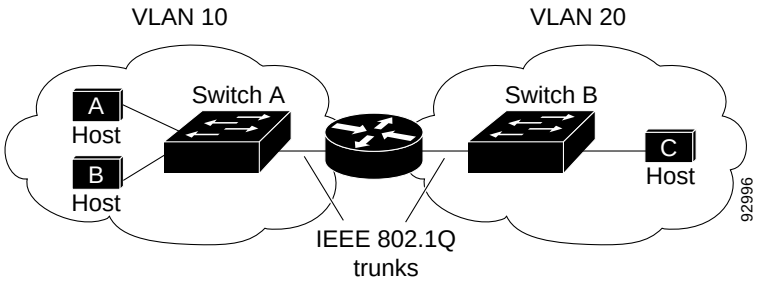
Note When configuring routing parameters on the switch to allocate system resources to maximize the number of unicast routes allowed, you should use the **sdm prefer default** global configuration command to set the Switch Database Management (sdm) feature to balance resource. The Layer-2 template does not support routing and forces any routing to be done through software. This overloads the CPU and severely degrades routing performance. For more information on the SDM templates, see Chapter 6, "Configuring SDM Templates" or see the **sdm prefer** command in the command reference for this release.

Understanding IP Routing

In some network environments, VLANs are associated with individual networks or subnetworks. In an IP network, each subnetwork is mapped to an individual VLAN. Configuring VLANs helps control the size of the broadcast domain and keeps local traffic local. However, network devices in different VLANs cannot communicate with one another without a Layer 3 device (router) to route traffic between the VLAN, referred to as inter-VLAN routing. You configure one or more routers to route traffic to the appropriate destination VLAN.

Figure 32-1 shows a basic routing topology. Switch A is in VLAN 10, and Switch B is in VLAN 20. The router has an interface in each VLAN.

Figure 32-1 Routing Topology Example



When Host A in VLAN 10 needs to communicate with Host B in VLAN 10, it sends a packet addressed to that host. Switch A forwards the packet directly to Host B, without sending it to the router.

When Host A sends a packet to Host C in VLAN 20, Switch A forwards the packet to the router, which receives the traffic on the VLAN 10 interface. The router checks the routing table, finds the correct outgoing interface, and forwards the packet on the VLAN 20 interface to Switch B. Switch B receives the packet and forwards it to Host C.

Types of Routing

Routers and Layer 3 switches can route packets in three different ways:

- By using default routing
- By using preprogrammed static routes for the traffic
- By dynamically calculating routes by using a routing protocol

Default routing refers to sending traffic with a destination unknown to the router to a default outlet or destination.

Static unicast routing forwards packets from predetermined ports through a single path into and out of a network. Static routing is secure and uses little bandwidth, but does not automatically respond to changes in the network, such as link failures, and therefore, might result in unreachable destinations. As networks grow, static routing becomes a labor-intensive liability.

Dynamic routing protocols are used by routers to dynamically calculate the best route for forwarding traffic. There are two types of dynamic routing protocols:

- Routers using distance-vector protocols maintain routing tables with distance values of networked resources, and periodically pass these tables to their neighbors. Distance-vector protocols use one or a series of metrics for calculating the best routes. These protocols are easy to configure and use.

- Routers using link-state protocols maintain a complex database of network topology, based on the exchange of link-state advertisements (LSAs) between routers. LSAs are triggered by an event in the network, which speeds up the convergence time or time required to respond to these changes. Link-state protocols respond quickly to topology changes, but require greater bandwidth and more resources than distance-vector protocols.

Distance-vector protocols supported by the switch are Routing Information Protocol (RIP), which uses a single distance metric (cost) to determine the best path and Border Gateway Protocol (BGP), which adds a path vector mechanism. The switch also supports the Open Shortest Path First (OSPF) link-state protocol and Enhanced IGRP (EIGRP), which adds some link-state routing features to traditional Interior Gateway Routing Protocol (IGRP) to improve efficiency.

Steps for Configuring Routing

By default, IPv4 routing is disabled on the switch, and you must enable it before routing can take place. For detailed IP routing configuration information, see the *Cisco IOS IP Configuration Guide, Release 12.2*

In the following procedures, the specified interface must be one of these Layer 3 interfaces:

- A routed port: a physical port configured as a Layer 3 port by using the **no switchport** interface configuration command.
- A switch virtual interface (SVI): a VLAN interface created by using the **interface vlan** *vlan_id* global configuration command and by default a Layer 3 interface.
- An EtherChannel port channel in Layer 3 mode: a port-channel logical interface created by using the **interface port-channel** *port-channel-number* global configuration command and binding the Ethernet interface into the channel group. For more information, see the "Configuring Layer 3 EtherChannels" section on page 31-13.

Note The switch does not support tunnel interfaces for unicast routed traffic.

All Layer 3 interfaces on which routing will occur must have IP addresses assigned to them. See the "Assigning IP Addresses to Network Interfaces" section on page 32-5.

Note A Layer 3 switch can have an IP address assigned to each routed port and SVI. The number of routed ports and SVIs that you can configure is not limited by software. However, the interrelationship between this number and the number and volume of features being implemented might have an impact on CPU utilization because of hardware limitations. To support routing, use the **sdm prefer default** global configuration command.

Configuring routing consists of several main procedures:

- To support VLAN interfaces, create and configure VLANs on the switch, and assign VLAN membership to Layer 2 interfaces. For more information, see Chapter 11, "Configuring VLANs."
- Configure Layer 3 interfaces.
- Enable IPv4 routing on the switch.
- Assign IPv4 addresses to the Layer 3 interfaces.
- Enable selected routing protocols on the switch.
- Configure routing protocol parameters (optional).

Configuring IP Addressing

A required task for configuring IP routing is to assign IP addresses to Layer 3 network interfaces to enable the interfaces and allow communication with the hosts on those interfaces that use IP. These sections describe how to configure various IP addressing features. Assigning IP addresses to the interface is required; the other procedures are optional.

- Default Addressing Configuration, page 32-4
- Assigning IP Addresses to Network Interfaces, page 32-5
- Configuring Address Resolution Methods, page 32-7
- Routing Assistance When IP Routing is Disabled, page 32-10
- Configuring Broadcast Packet Handling, page 32-12
- Monitoring and Maintaining IP Addressing, page 32-17

Default Addressing Configuration

Table 32-1 shows the default addressing configuration.

Table 32-1 Default Addressing Configuration

Feature	Default Setting
IP address	None defined.
ARP	No permanent entries in the Address Resolution Protocol (ARP) cache. Encapsulation: Standard Ethernet-style ARP. Timeout: 14400 seconds (4 hours).
IP broadcast address	255.255.255.255 (all ones).
IP classless routing	Enabled.
IP default gateway	Disabled.
IP directed broadcast	Disabled (all IP directed broadcasts are dropped).
IP domain	Domain list: No domain names defined. Domain lookup: Enabled. Domain name: Enabled.
IP forward-protocol	If a helper address is defined or User Datagram Protocol (UDP) flooding is configured, UDP forwarding is enabled on default ports. Any-local-broadcast: Disabled. Turbo-flood: Disabled.
IP helper address	Disabled.
IP host	Disabled.

Table 32-1 Default Addressing Configuration (continued)

Feature	Default Setting
IRDP	Disabled. Defaults when enabled: • Broadcast IRDP advertisements. • Maximum interval between advertisements: 600 seconds. • Minimum interval between advertisements: 0.75 times max interval • Preference: 0.
IP proxy ARP	Enabled.
IP routing	Disabled.
IP subnet-zero	Disabled.

Assigning IP Addresses to Network Interfaces

An IP address identifies a location to which IP packets can be sent. Some IP addresses are reserved for special uses and cannot be used for host, subnet, or network addresses. RFC 1166, “Internet Numbers,” contains the official description of IP addresses.

An interface can have one primary IP address. A mask identifies the bits that denote the network number in an IP address. When you use the mask to subnet a network, the mask is referred to as a subnet mask. To receive an assigned network number, contact your Internet service provider.

Beginning in privileged EXEC mode, follow these steps to assign an IP address and a network mask to a Layer 3 interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and specify the Layer 3 interface to configure.
Step 3	**no shutdown**	Enable the interface if necessary. User network interfaces (UNIs) are disabled by default; network node interfaces (NNIs) are enabled by default.
Step 4	**no switchport**	Remove the interface from Layer 2 configuration mode (if it is a physical interface).
Step 5	**ip address** *ip-address subnet-mask*	Configure the IP address and IP subnet mask.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show interfaces** [*interface-id*] **show ip interface** [*interface-id*] **show running-config interface** [*interface-id*]	Verify your entries.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use of Subnet Zero

Subnetting with a subnet address of zero is strongly discouraged because of the problems that can arise if a network and a subnet have the same addresses. For example, if network 131.108.0.0 is subnetted as 255.255.255.0, subnet zero would be written as 131.108.0.0, which is the same as the network address.

You can use the all ones subnet (131.108.255.0) and even though it is discouraged, you can enable the use of subnet zero if you need the entire subnet space for your IP address.

Beginning in privileged EXEC mode, follow these steps to enable subnet zero:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip subnet-zero**	Enable the use of subnet zero for interface addresses and routing updates.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entry.
Step 5	**copy running-config startup-config**	(Optional) Save your entry in the configuration file.

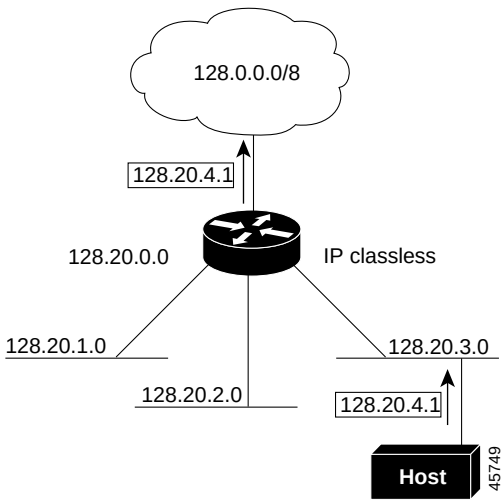
Use the **no ip subnet-zero** global configuration command to restore the default and disable the use of subnet zero.

Classless Routing

By default, classless routing behavior is enabled on the switch when it is configured to route. With classless routing, if a router receives packets for a subnet of a network with no default route, the router forwards the packet to the best supernet route. A *supernet* consists of contiguous blocks of Class C address spaces used to simulate a single, larger address space and is designed to relieve the pressure on the rapidly depleting Class B address space.

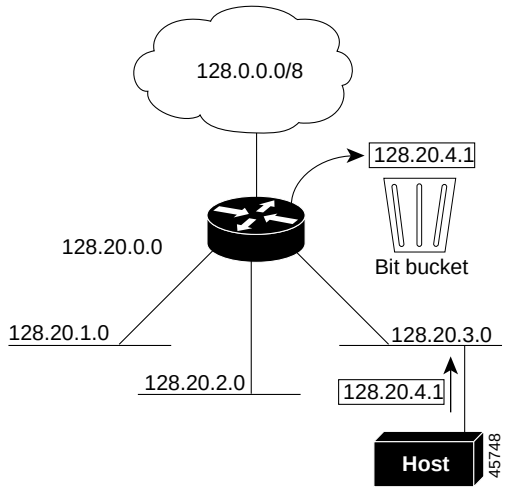
In Figure 32-2, classless routing is enabled. When the host sends a packet to 120.20.4.1, instead of discarding the packet, the router forwards it to the best supernet route. If you disable classless routing and a router receives packets destined for a subnet of a network with no network default route, the router discards the packet.

Figure 32-2 IP Classless Routing



In Figure 32-3, the router in network 128.20.0.0 is connected to subnets 128.20.1.0, 128.20.2.0, and 128.20.3.0. If the host sends a packet to 120.20.4.1, because there is no network default route, the router discards the packet.

Figure 32-3 No IP Classless Routing



To prevent the switch from forwarding packets destined for unrecognized subnets to the best supernet route possible, you can disable classless routing behavior.

Beginning in privileged EXEC mode, follow these steps to disable classless routing:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**no ip classless**	Disable classless routing behavior.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entry.
Step 5	**copy running-config startup-config**	(Optional) Save your entry in the configuration file.

To restore the default and have the switch forward packets destined for a subnet of a network with no network default route to the best supernet route possible, use the **ip classless** global configuration command.

Configuring Address Resolution Methods

You can control interface-specific handling of IP by using address resolution. A device using IP can have both a local address or MAC address, which uniquely defines the device on its local segment or LAN, and a network address, which identifies the network to which the device belongs.

The local address or MAC address is known as a data link address because it is contained in the data link layer (Layer 2) section of the packet header and is read by data link (Layer 2) devices. To communicate with a device on Ethernet, the software must learn the MAC address of the device. The process of learning the MAC address from an IP address is called *address resolution*. The process of learning the IP address from the MAC address is called *reverse address resolution*.

The switch can use these forms of address resolution:

- Address Resolution Protocol (ARP) is used to associate IP address with MAC addresses. Taking an IP address as input, ARP learns the associated MAC address and then stores the IP address/MAC address association in an ARP cache for rapid retrieval. Then the IP datagram is encapsulated in a link-layer frame and sent over the network. Encapsulation of IP datagrams and ARP requests or replies on IEEE 802 networks other than Ethernet is specified by the Subnetwork Access Protocol (SNAP).
- Proxy ARP helps hosts with no routing tables learn the MAC addresses of hosts on other networks or subnets. If the switch (router) receives an ARP request for a host that is not on the same interface as the ARP request sender, and if the router has all of its routes to the host through other interfaces, it generates a proxy ARP packet giving its own local data link address. The host that sent the ARP request then sends its packets to the router, which forwards them to the intended host.

The switch also uses the Reverse Address Resolution Protocol (RARP), which functions the same as ARP does, except that the RARP packets request an IP address instead of a local MAC address. Using RARP requires a RARP server on the same network segment as the router interface. Use the **ip rarp-server** *address* interface configuration command to identify the server.

For more information on RARP, see the *Cisco IOS Configuration Fundamentals Configuration Guide, Release 12.2*.

You can perform these tasks to configure address resolution:

- Define a Static ARP Cache, page 32-8
- Set ARP Encapsulation, page 32-9
- Enable Proxy ARP, page 32-10

Define a Static ARP Cache

ARP and other address resolution protocols provide dynamic mapping between IP addresses and MAC addresses. Because most hosts support dynamic address resolution, you usually do not need to specify static ARP cache entries. If you must define a static ARP cache entry, you can do so globally, which installs a permanent entry in the ARP cache that the switch uses to translate IP addresses into MAC addresses. Optionally, you can also specify that the switch respond to ARP requests as if it were the owner of the specified IP address. If you do not want the ARP entry to be permanent, you can specify a timeout period for the ARP entry.

Beginning in privileged EXEC mode, follow these steps to provide static mapping between IP addresses and MAC addresses:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**arp** *ip-address hardware-address type*	Globally associate an IP address with a MAC (hardware) address in the ARP cache, and specify encapsulation type as one of these: • **arpa**—ARP encapsulation for Ethernet interfaces • **snap**—Subnetwork Address Protocol encapsulation for Token Ring and FDDI interfaces • **sap**—HP's ARP type

	Command	Purpose
Step 3	**arp** *ip-address hardware-address type* [**alias**]	(Optional) Specify that the switch respond to ARP requests as if it were the owner of the specified IP address.
Step 4	**interface** *interface-id*	Enter interface configuration mode, and specify the interface to configure.
Step 5	**no shutdown**	Enable the interface if necessary. By default, UNIs are disabled and NNIs are enabled.
Step 6	**arp timeout** *seconds*	(Optional) Set the length of time an ARP cache entry will stay in the cache. The default is 14400 seconds (4 hours). The range is 0 to 2147483 seconds.
Step 7	**end**	Return to privileged EXEC mode.
Step 8	**show interfaces** [*interface-id*]	Verify the type of ARP and the timeout value used on all interfaces or a specific interface.
Step 9	**show arp** or **show ip arp**	View the contents of the ARP cache.
Step 10	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove an entry from the ARP cache, use the **no arp** *ip-address hardware-address type* global configuration command. To remove all nonstatic entries from the ARP cache, use the **clear arp-cache** privileged EXEC command.

Set ARP Encapsulation

By default, Ethernet ARP encapsulation (represented by the **arpa** keyword) is enabled on an IP interface. You can change the encapsulation methods to SNAP if required by your network.

Beginning in privileged EXEC mode, follow these steps to specify the ARP encapsulation type:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and specify the Layer 3 interface to configure.
Step 3	**no shutdown**	Enable the interface if necessary. By default, UNIs are disabled and NNIs are enabled.
Step 4	**arp** {**arpa** \| **snap**}	Specify the ARP encapsulation method: • **arpa**—Address Resolution Protocol • **snap**—Subnetwork Address Protocol
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show interfaces** [*interface-id*]	Verify ARP encapsulation configuration on all interfaces or the specified interface.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable an encapsulation type, use the **no arp arpa** or **no arp snap** interface configuration command.

Enable Proxy ARP

By default, the switch uses proxy ARP to help hosts learn MAC addresses of hosts on other networks or subnets.

Beginning in privileged EXEC mode, follow these steps to enable proxy ARP if it has been disabled:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and specify the Layer 3 interface to configure.
Step 3	**no shutdown**	Enable the interface if necessary. By default, UNIs are disabled and NNIs are enabled.
Step 4	**ip proxy-arp**	Enable proxy ARP on the interface.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show ip interface** [*interface-id*]	Verify the configuration on the interface or all interfaces.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable proxy ARP on the interface, use the **no ip proxy-arp** interface configuration command.

Routing Assistance When IP Routing is Disabled

These mechanisms allow the switch to learn about routes to other networks when it does not have IP routing enabled:

- Proxy ARP, page 32-10
- Default Gateway, page 32-11
- ICMP Router Discovery Protocol (IRDP), page 32-11

Proxy ARP

Proxy ARP, the most common method for learning about other routes, enables an Ethernet host with no routing information to communicate with hosts on other networks or subnets. The host assumes that all hosts are on the same local Ethernet and that they can use ARP to learn their MAC addresses. If a switch receives an ARP request for a host that is not on the same network as the sender, the switch evaluates whether it has the best route to that host. If it does, it sends an ARP reply packet with its own Ethernet MAC address, and the host that sent the request sends the packet to the switch, which forwards it to the intended host. Proxy ARP treats all networks as if they are local and performs ARP requests for every IP address.

Proxy ARP is enabled by default. To enable it after it has been disabled, see the "Enable Proxy ARP" section on page 32-10. Proxy ARP works as long as other routers support it.

Default Gateway

Another method for locating routes is to define a default router or default gateway. All nonlocal packets are sent to this router, which either routes them appropriately or sends an IP Control Message Protocol (ICMP) redirect message back, defining which local router the host should use. The switch caches the redirect messages and forwards each packet as efficiently as possible. A limitation of this method is that there is no means of detecting when the default router has gone down or is unavailable.

Beginning in privileged EXEC mode, follow these steps to define a default gateway (router) when IP routing is disabled:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip default-gateway** *ip-address*	Set up a default gateway (router).
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show ip redirects**	Display the address of the default gateway router to verify the setting.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no ip default-gateway** global configuration command to disable this function.

ICMP Router Discovery Protocol (IRDP)

Router discovery allows the switch to dynamically learn about routes to other networks using IRDP. IRDP allows hosts to locate routers. When operating as a client, the switch generates router discovery packets. When operating as a host, the switch receives router discovery packets. The switch can also listen to Routing Information Protocol (RIP) routing updates and use this information to infer locations of routers. The switch does not actually store the routing tables sent by routing devices; it merely keeps track of which systems are sending the data. The advantage of using IRDP is that it allows each router to specify both a priority and the time after which a device is assumed to be down if no further packets are received.

Each device discovered becomes a candidate for the default router, and a new highest-priority router is selected when a higher priority router is discovered, when the current default router is declared down, or when a TCP connection is about to time out because of excessive retransmissions.

The only required task for IRDP routing on an interface is to enable IRDP processing on that interface. When enabled, the default parameters apply. You can optionally change any of these parameters.

Beginning in privileged EXEC mode, follow these steps to enable and configure IRDP on an interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and specify the Layer 3 interface to configure.
Step 3	**no shutdown**	Enable the interface if necessary. By default, UNIs are disabled and NNIs are enabled.
Step 4	**ip irdp**	Enable IRDP processing on the interface.

	Command	Purpose
Step 5	**ip irdp multicast**	(Optional) Send IRDP advertisements to the multicast address (224.0.0.1) instead of IP broadcasts. **Note** This command allows for compatibility with Sun Microsystems Solaris, which requires IRDP packets to be sent out as multicasts. Many implementations cannot receive these multicasts; ensure end-host ability before using this command.
Step 6	**ip irdp holdtime** *seconds*	(Optional) Set the IRDP period for which advertisements are valid. The default is three times the **maxadvertinterval** value. It must be greater than **maxadvertinterval** and cannot be greater than 9000 seconds. If you change the **maxadvertinterval** value, this value also changes.
Step 7	**ip irdp maxadvertinterval** *seconds*	(Optional) Set the IRDP maximum interval between advertisements. The default is 600 seconds.
Step 8	**ip irdp minadvertinterval** *seconds*	(Optional) Set the IRDP minimum interval between advertisements. The default is 0.75 times the **maxadvertinterval**. If you change the **maxadvertinterval**, this value changes to the new default (0.75 of **maxadvertinterval**).
Step 9	**ip irdp preference** *number*	(Optional) Set a device IRDP preference level. The allowed range is -2^{31} to 2^{31}. The default is 0. A higher value increases the router preference level.
Step 10	**ip irdp address** *address* [*number*]	(Optional) Specify an IRDP address and preference to proxy-advertise.
Step 11	**end**	Return to privileged EXEC mode.
Step 12	**show ip irdp**	Verify settings by displaying IRDP values.
Step 13	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

If you change the **maxadvertinterval** value, the **holdtime** and **minadvertinterval** values also change, so it is important to first change the **maxadvertinterval** value, before manually changing either the **holdtime** or **minadvertinterval** values.

Use the **no ip irdp** interface configuration command to disable IRDP routing.

Configuring Broadcast Packet Handling

After configuring an IP interface address, you can enable routing and configure one or more routing protocols, or you can configure the way the switch responds to network broadcasts. A broadcast is a data packet destined for all hosts on a physical network. The switch supports two kinds of broadcasting:

- A directed broadcast packet is sent to a specific network or series of networks. A directed broadcast address includes the network or subnet fields.
- A flooded broadcast packet is sent to every network.

Note You can also limit broadcast, unicast, and multicast traffic on Layer 2 interfaces by using the **storm-control** interface configuration command to set traffic suppression levels. For more information, see Chapter 21, “Configuring Port-Based Traffic Control.”

Routers provide some protection from broadcast storms by limiting their extent to the local cable. Bridges (including intelligent bridges), because they are Layer 2 devices, forward broadcasts to all network segments, thus propagating broadcast storms. The best solution to the broadcast storm problem is to use a single broadcast address scheme on a network. In most modern IP implementations, you can set the address to be used as the broadcast address. Many implementations, including the one in the switch, support several addressing schemes for forwarding broadcast messages.

Perform the tasks in these sections to enable these schemes:

- Enabling Directed Broadcast-to-Physical Broadcast Translation, page 32-13
- Forwarding UDP Broadcast Packets and Protocols, page 32-14
- Establishing an IP Broadcast Address, page 32-15
- Flooding IP Broadcasts, page 32-15

Enabling Directed Broadcast-to-Physical Broadcast Translation

By default, IP directed broadcasts are dropped; they are not forwarded. Dropping IP-directed broadcasts makes routers less susceptible to denial-of-service attacks.

You can enable forwarding of IP-directed broadcasts on an interface where the broadcast becomes a physical (MAC-layer) broadcast. Only those protocols configured by using the **ip forward-protocol** global configuration command are forwarded.

You can specify an access list to control which broadcasts are forwarded. When an access list is specified, only those IP packets permitted by the access list are eligible to be translated from directed broadcasts to physical broadcasts. For more information on access lists, see Chapter 28, “Configuring Network Security with ACLs.”

Beginning in privileged EXEC mode, follow these steps to enable forwarding of IP-directed broadcasts on an interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and specify the interface to configure.
Step 3	**no shutdown**	Enable the interface if necessary. By default, UNIs are disabled and NNIs are enabled.
Step 4	**ip directed-broadcast** [*access-list-number*]	Enable directed broadcast-to-physical broadcast translation on the interface. You can include an access list to control which broadcasts are forwarded. When an access list is specified, only IP packets permitted by the access list are eligible to be translated.
Step 5	**exit**	Return to global configuration mode.
Step 6	**ip forward-protocol** {**udp** [*port*] \| **nd** \| **sdns**}	Specify which protocols and ports the router forwards when forwarding broadcast packets. • **udp**—Forward UPD datagrams. *port*: (Optional) Destination port that controls which UDP services are forwarded. • **nd**—Forward ND datagrams. • **sdns**—Forward SDNS datagrams

	Command	Purpose
Step 7	**end**	Return to privileged EXEC mode.
Step 8	**show ip interface** [*interface-id*] or **show running-config**	Verify the configuration on the interface or all interfaces.
Step 9	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no ip directed-broadcast** interface configuration command to disable translation of directed broadcast to physical broadcasts. Use the **no ip forward-protocol** global configuration command to remove a protocol or port.

Forwarding UDP Broadcast Packets and Protocols

User Datagram Protocol (UDP) is an IP host-to-host layer protocol, as is TCP. UDP provides a low-overhead, connectionless session between two end systems and does not provide for acknowledgment of received datagrams. Network hosts occasionally use UDP broadcasts to find address, configuration, and name information. If such a host is on a network segment that does not include a server, UDP broadcasts are normally not forwarded. You can remedy this situation by configuring an interface on a router to forward certain classes of broadcasts to a helper address. You can use more than one helper address per interface.

You can specify a UDP destination port to control which UDP services are forwarded. You can specify multiple UDP protocols. You can also specify the Network Disk (ND) protocol, which is used by older diskless Sun workstations and the network security protocol SDNS.

By default, both UDP and ND forwarding are enabled if a helper address has been defined for an interface. The description for the **ip forward-protocol** interface configuration command in the *Cisco IOS IP Command Reference, Volume 1 of 3: Addressing and Services, Release 12.2* lists the ports that are forwarded by default if you do not specify any UDP ports.

If you do not specify any UDP ports when you configure the forwarding of UDP broadcasts, you are configuring the router to act as a BOOTP forwarding agent. BOOTP packets carry DHCP information.

Beginning in privileged EXEC mode, follow these steps to enable forwarding UDP broadcast packets on an interface and specify the destination address:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and specify the Layer 3 interface to configure.
Step 3	**no shutdown**	Enable the interface if necessary. By default, UNIs are disabled and NNIs are enabled.
Step 4	**ip helper-address** *address*	Enable forwarding and specify the destination address for forwarding UDP broadcast packets, including BOOTP.
Step 5	**exit**	Return to global configuration mode.
Step 6	**ip forward-protocol** {**udp** [*port*] \| **nd** \| **sdns**}	Specify which protocols the router forwards when forwarding broadcast packets.
Step 7	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 8	**show ip interface** [*interface-id*] or **show running-config**	Verify the configuration on the interface or all interfaces.
Step 9	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no ip helper-address** interface configuration command to disable the forwarding of broadcast packets to specific addresses. Use the **no ip forward-protocol** global configuration command to remove a protocol or port.

Establishing an IP Broadcast Address

The most popular IP broadcast address (and the default) is an address consisting of all ones (255.255.255.255). However, the switch can be configured to generate any form of IP broadcast address.

Beginning in privileged EXEC mode, follow these steps to set the IP broadcast address on an interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and specify the interface to configure.
Step 3	**no shutdown**	Enable the interface if necessary. By default, UNIs are disabled and NNIs are enabled.
Step 4	**ip broadcast-address** *ip-address*	Enter a broadcast address different from the default, for example 128.1.255.255.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show ip interface** [*interface-id*]	Verify the broadcast address on the interface or all interfaces.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To restore the default IP broadcast address, use the **no ip broadcast-address** interface configuration command.

Flooding IP Broadcasts

You can allow IP broadcasts to be flooded throughout your internetwork in a controlled fashion by using the database created by the bridging STP. Using this feature also prevents loops. To support this capability, bridging must be configured on each interface that is to participate in the flooding. If bridging is not configured on an interface, it still can receive broadcasts. However, the interface never forwards broadcasts it receives, and the router never uses that interface to send broadcasts received on a different interface.

Packets that are forwarded to a single network address using the IP helper-address mechanism can be flooded. Only one copy of the packet is sent on each network segment.

To be considered for flooding, packets must meet these criteria. (Note that these are the same conditions used to consider packet forwarding using IP helper addresses.)

- The packet must be a MAC-level broadcast.
- The packet must be an IP-level broadcast.
- The packet must be a TFTP, DNS, Time, NetBIOS, ND, or BOOTP packet, or a UDP specified by the **ip forward-protocol udp** global configuration command.
- The time-to-live (TTL) value of the packet must be at least two.

A flooded UDP datagram is given the destination address specified with the **ip broadcast-address** interface configuration command on the output interface. The destination address can be set to any address. Thus, the destination address might change as the datagram propagates through the network. The source address is never changed. The TTL value is decremented.

When a flooded UDP datagram is sent out an interface (and the destination address possibly changed), the datagram is handed to the normal IP output routines and is, therefore, subject to access lists, if they are present on the output interface.

Beginning in privileged EXEC mode, follow these steps to use the bridging spanning-tree database to flood UDP datagrams:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip forward-protocol spanning-tree**	Use the bridging spanning-tree database to flood UDP datagrams.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entry.
Step 5	**copy running-config startup-config**	(Optional) Save your entry in the configuration file.

Use the **no ip forward-protocol spanning-tree** global configuration command to disable the flooding of IP broadcasts.

In the switch, the majority of packets are forwarded in hardware; most packets do not go through the switch CPU. For those packets that do go to the CPU, you can speed up spanning tree-based UDP flooding by a factor of about four to five times by using turbo-flooding. This feature is supported over Ethernet interfaces configured for ARP encapsulation.

Beginning in privileged EXEC mode, follow these steps to increase spanning-tree-based flooding:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode
Step 2	**ip forward-protocol turbo-flood**	Use the spanning-tree database to speed up flooding of UDP datagrams.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entry.
Step 5	**copy running-config startup-config**	(Optional) Save your entry in the configuration file.

To disable this feature, use the **no ip forward-protocol turbo-flood** global configuration command.

Monitoring and Maintaining IP Addressing

When the contents of a particular cache, table, or database have become or are suspected to be invalid, you can remove all its contents by using the **clear** privileged EXEC commands. Table 32-2 lists the commands for clearing contents.

Table 32-2 Commands to Clear Caches, Tables, and Databases

Command	Purpose
clear arp-cache	Clear the IP ARP cache and the fast-switching cache.
clear host {*name* \| *}}	Remove one or all entries from the hostname and the address cache.
clear ip route {*network* [*mask*] \|*}	Remove one or more routes from the IP routing table.

You can display specific statistics, such as the contents of IP routing tables, caches, and databases; the reachability of nodes; and the routing path that packets are taking through the network. Table 32-3 lists the privileged EXEC commands for displaying IP statistics.

Table 32-3 Commands to Display Caches, Tables, and Databases

Command	Purpose
show arp	Display the entries in the ARP table.
show hosts	Display the default domain name, style of lookup service, name server hosts, and the cached list of hostnames and addresses.
show ip aliases	Display IP addresses mapped to TCP ports (aliases).
show ip arp	Display the IP ARP cache.
show ip interface [*interface-id*]	Display the IP status of interfaces.
show ip irdp	Display IRDP values.
show ip masks *address*	Display the masks used for network addresses and the number of subnets using each mask.
show ip redirects	Display the address of a default gateway.
show ip route [*address* [*mask*]] \| [*protocol*]	Display the current state of the routing table.
show ip route summary	Display the current state of the routing table in summary form.

Enabling IPv4 Unicast Routing

By default, the switch is in Layer 2 switching mode and IP routing is disabled. To use the Layer 3 capabilities of the switch, you must enable IP routing.

Beginning in privileged EXEC mode, follow these steps to enable IP routing:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip routing**	Enable IP routing.

	Command	Purpose
Step 3	**router** *ip_routing_protocol*	Specify an IP routing protocol. This step might include other commands, such as specifying the networks to route with the **network** (RIP) router configuration command. For information on specific protocols, see sections later in this chapter and to the *Cisco IOS IP Configuration Guide, Release 12.2*.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no ip routing** global configuration command to disable routing.

This example shows how to enable IP routing using RIP as the routing protocol:

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# ip routing
Switch(config)# router rip
Switch(config-router)# network 10.0.0.0
Switch(config-router)# end
```

You can now set up parameters for the selected routing protocols as described in these sections:

- Configuring RIP, page 32-18
- Configuring OSPF, page 32-23
- Configuring EIGRP, page 32-32
- Configuring BGP, page 32-38
- Configuring Protocol-Independent Features, page 32-69 (optional)

Configuring RIP

The Routing Information Protocol (RIP) is an interior gateway protocol (IGP) created for use in small, homogeneous networks. It is a distance-vector routing protocol that uses broadcast User Datagram Protocol (UDP) data packets to exchange routing information. The protocol is documented in RFC 1058. You can find detailed information about RIP in *IP Routing Fundamentals*, published by Cisco Press.

Using RIP, the switch sends routing information updates (advertisements) every 30 seconds. If a router does not receive an update from another router for 180 seconds or more, it marks the routes served by that router as unusable. If there is still no update after 240 seconds, the router removes all routing table entries for the non-updating router.

RIP uses hop counts to rate the value of different routes. The hop count is the number of routers that can be traversed in a route. A directly connected network has a hop count of zero; a network with a hop count of 16 is unreachable. This small range (0 to 15) makes RIP unsuitable for large networks.

If the router has a default network path, RIP advertises a route that links the router to the pseudonetwork 0.0.0.0. The 0.0.0.0 network does not exist; it is treated by RIP as a network to implement the default routing feature. The switch advertises the default network if a default was learned by RIP or if the router has a gateway of last resort and RIP is configured with a default metric. RIP sends updates to the interfaces in specified networks. If an interface's network is not specified, it is not advertised in any RIP update.

These sections contain this configuration information:

- Default RIP Configuration, page 32-19
- Configuring Basic RIP Parameters, page 32-19
- Configuring RIP Authentication, page 32-21
- Configuring Summary Addresses and Split Horizon, page 32-21

Default RIP Configuration

Table 32-4 shows the default RIP configuration.

Table 32-4 Default RIP Configuration

Feature	Default Setting
Auto summary	Enabled.
Default-information originate	Disabled.
Default metric	Built-in; automatic metric translations.
IP RIP authentication key-chain	No authentication. Authentication mode: clear text.
IP RIP receive version	According to the **version** router configuration command.
IP RIP send version	According to the **version** router configuration command.
IP RIP triggered	According to the **version** router configuration command.
IP split horizon	Varies with media.
Neighbor	None defined.
Network	None specified.
Offset list	Disabled.
Output delay	0 milliseconds.
Timers basic	• Update: 30 seconds. • Invalid: 180 seconds. • Hold-down: 180 seconds. • Flush: 240 seconds.
Validate-update-source	Enabled.
Version	Receives RIP Version 1 and 2 packets; sends Version 1 packets.

Configuring Basic RIP Parameters

To configure RIP, you enable RIP routing for a network and optionally configure other parameters.

Beginning in privileged EXEC mode, follow these steps to enable and configure RIP:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip routing**	Enable IP routing. (Required only if IP routing is disabled.)
Step 3	**router rip**	Enable a RIP routing process, and enter router configuration mode.
Step 4	**network** *network number*	Associate a network with a RIP routing process. You can specify multiple **network** commands. RIP routing updates are sent and received through interfaces only on these networks.
Step 5	**neighbor** *ip-address*	(Optional) Define a neighboring router with which to exchange routing information. This step allows routing updates from RIP (normally a broadcast protocol) to reach nonbroadcast networks.
Step 6	**offset list** [*access-list number* \| *name*] {**in** \| **out**} *offset* [*type number*]	(Optional) Apply an offset list to routing metrics to increase incoming and outgoing metrics to routes learned through RIP. You can limit the offset list with an access list or an interface.
Step 7	**timers basic** *update invalid holddown flush*	(Optional) Adjust routing protocol timers. Valid ranges for all timers are 0 to 4294967295 seconds. • *update*—The time between sending routing updates. The default is 30 seconds. • *invalid*—The timer after which a route is declared invalid. The default is 180 seconds. • *holddown*—The time before a route is removed from the routing table. The default is 180 seconds. • *flush*—The amount of time for which routing updates are postponed. The default is 240 seconds.
Step 8	**version** {**1** \| **2**}	(Optional) Configure the switch to receive and send only RIP Version 1 or RIP Version 2 packets. By default, the switch receives Version 1 and 2 but sends only Version 1. You can also use the interface commands **ip rip** {**send** \| **receive**} **version 1** \| **2** \| **1 2**} to control what versions are used for sending and receiving on interfaces.
Step 9	**no auto summary**	(Optional) Disable automatic summarization. By default, the switch summarizes subprefixes when crossing classful network boundaries. Disable summarization (RIP Version 2 only) to advertise subnet and host routing information to classful network boundaries.
Step 10	**no validate-update-source**	(Optional) Disable validation of the source IP address of incoming RIP routing updates. By default, the switch validates the source IP address of incoming RIP routing updates and discards the update if the source address is not valid. Under normal circumstances, disabling this feature is not recommended. However, if you have a router that is off-network and you want to receive its updates, you can use this command.
Step 11	**output-delay** *delay*	(Optional) Add interpacket delay for RIP updates sent. By default, packets in a multiple-packet RIP update have no delay added between packets. If you are sending packets to a lower-speed device, you can add an interpacket delay in the range of 8 to 50 milliseconds.
Step 12	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 13	**show ip protocols**	Verify your entries.
Step 14	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To turn off the RIP routing process, use the **no router rip** global configuration command.

To display the parameters and current state of the active routing protocol process, use the **show ip protocols** privileged EXEC command. Use the **show ip rip database** privileged EXEC command to display summary address entries in the RIP database.

Configuring RIP Authentication

RIP Version 1 does not support authentication. If you are sending and receiving RIP Version 2 packets, you can enable RIP authentication on an interface. The key chain specifies \the set of keys that can be used on the interface. If a key chain is not configured, no authentication is performed, not even the default. Therefore, you must also perform the tasks in the "Managing Authentication Keys" section on page 32-82.

The switch supports two modes of authentication on interfaces for which RIP authentication is enabled: plain text and MD5. The default is plain text.

Beginning in privileged EXEC mode, follow these steps to configure RIP authentication on an interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and specify the interface to configure.
Step 3	**no shutdown**	Enable the interface if necessary. By default, UNIs are disabled and NNIs are enabled.
Step 4	**ip rip authentication key-chain** *name-of-chain*	Enable RIP authentication.
Step 5	**ip rip authentication mode** [**text** \| **md5**}	Configure the interface to use plain text authentication (the default) or MD5 digest authentication.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show running-config interface** [*interface-id*]	Verify your entries.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To restore clear text authentication, use the **no ip rip authentication mode** interface configuration command. To prevent authentication, use the **no ip rip authentication key-chain** interface configuration command.

Configuring Summary Addresses and Split Horizon

Routers connected to broadcast-type IP networks and using distance-vector routing protocols normally use the split-horizon mechanism to reduce the possibility of routing loops. Split horizon blocks information about routes from being advertised by a router on any interface from which that information originated. This feature usually optimizes communication among multiple routers, especially when links are broken.

Note In general, disabling split horizon is not recommended unless you are certain that your application requires it to properly advertise routes.

If you want to configure an interface running RIP to advertise a summarized local IP address pool on a network access server for dial-up clients, use the **ip summary-address rip** interface configuration command.

Note If split horizon is enabled, neither autosummary nor interface IP summary addresses are advertised.

Beginning in privileged EXEC mode, follow these steps to set an interface to advertise a summarized local IP address and to disable split horizon on the interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and specify the Layer 3 interface to configure.
Step 3	**no shutdown**	Enable the interface if necessary. By default, UNIs are disabled and NNIs are enabled.
Step 4	**ip address** *ip-address subnet-mask*	Configure the IP address and IP subnet.
Step 5	**ip summary-address rip** *ip address ip-network mask*	Configure the IP address to be summarized and the IP network mask.
Step 6	**no ip split horizon**	Disable split horizon on the interface.
Step 7	**end**	Return to privileged EXEC mode.
Step 8	**show ip interface** *interface-id*	Verify your entries.
Step 9	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable IP summarization, use the **no ip summary-address rip** router configuration command.

In this example, the major net is 10.0.0.0. The summary address 10.2.0.0 overrides the autosummary address of 10.0.0.0 so that 10.2.0.0 is advertised out interface Gigabit Ethernet port 2, and 10.0.0.0 is not advertised. In the example, if the interface is still in Layer 2 mode (the default), you must enter a **no switchport** interface configuration command before entering the **ip address** interface configuration command.

Note If split horizon is enabled, neither autosummary nor interface summary addresses (those configured with the **ip summary-address rip** router configuration command) are advertised.

```
Switch(config)# router rip
Switch(config-router)# interface gi0/2
Switch(config-if)# ip address 10.1.5.1 255.255.255.0
Switch(config-if)# ip summary-address rip 10.2.0.0 255.255.0.0
Switch(config-if)# no ip split-horizon
Switch(config-if)# exit
Switch(config)# router rip
Switch(config-router)# network 10.0.0.0
Switch(config-router)# neighbor 2.2.2.2 peer-group mygroup
Switch(config-router)# end
```

Configuring Split Horizon

Routers connected to broadcast-type IP networks and using distance-vector routing protocols normally use the split-horizon mechanism to reduce the possibility of routing loops. Split horizon blocks information about routes from being advertised by a router on any interface from which that information originated. This feature can optimize communication among multiple routers, especially when links are broken.

Note In general, Cisco does not recommend disabling split horizon unless you are certain that your application requires it to properly advertise routes.

Beginning in privileged EXEC mode, follow these steps to disable split horizon on the interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and specify the interface to configure.
Step 3	**no shutdown**	Enable the interface if necessary. By default, UNIs are disabled and NNIs are enabled.
Step 4	**ip address** *ip-address subnet-mask*	Configure the IP address and IP subnet.
Step 5	**no ip split-horizon**	Disable split horizon on the interface.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show ip interface** *interface-id*	Verify your entries.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To enable the split horizon mechanism, use the **ip split-horizon** interface configuration command.

Configuring OSPF

This section briefly describes how to configure Open Shortest Path First (OSPF). For a complete description of the OSPF commands, see the "OSPF Commands" chapter of the *Cisco IOS IP Command Reference, Volume 2 of 3: Routing Protocols, Release 12.2*.

Note OSPF classifies different media into broadcast, nonbroadcast, and point-to-point networks. The switch supports broadcast and point-to-point networks (Ethernet interfaces configured as point-to-point links).

OSPF is an Interior Gateway Protocol (IGP) designed expressly for IP networks, supporting IP subnetting and tagging of externally derived routing information. OSPF also allows packet authentication and uses IP multicast when sending and receiving packets. The Cisco implementation supports RFC 1253, OSPF management information base (MIB).

The Cisco implementation conforms to the OSPF Version 2 specifications with these key features:

- Definition of stub areas is supported.
- Routes learned through any IP routing protocol can be redistributed into another IP routing protocol. At the intradomain level, this means that OSPF can import routes learned through EIGRP and RIP. OSPF routes can also be exported into RIP.
- Plain text and MD5 authentication among neighboring routers within an area is supported.
- Configurable routing interface parameters include interface output cost, retransmission interval, interface transmit delay, router priority, router dead and hello intervals, and authentication key.
- Virtual links are supported.
- Not-so-stubby-areas (NSSAs) per RFC 1587are supported.

OSPF typically requires coordination among many internal routers, *area border routers* (ABRs) connected to multiple areas, and *autonomous system boundary routers* (ASBRs). The minimum configuration would use all default parameter values, no authentication, and interfaces assigned to areas. If you customize your environment, you must ensure coordinated configuration of all routers.

These sections contain this configuration information:

- Default OSPF Configuration, page 32-24
- Configuring Basic OSPF Parameters, page 32-26
- Configuring OSPF Interfaces, page 32-26
- Configuring OSPF Area Parameters, page 32-27
- Configuring Other OSPF Parameters, page 32-29
- Changing LSA Group Pacing, page 32-30
- Configuring a Loopback Interface, page 32-31
- Monitoring OSPF, page 32-31

Default OSPF Configuration

Table 32-5 shows the default OSPF configuration.

Table 32-5 Default OSPF Configuration

Feature	Default Setting
Interface parameters	Cost: No default cost predefined.
	Retransmit interval: 5 seconds.
	Transmit delay: 1 second.
	Priority: 1.
	Hello interval: 10 seconds.
	Dead interval: 4 times the hello interval.
	No authentication.
	No password specified.
	MD5 authentication disabled.

Table 32-5 Default OSPF Configuration (continued)

Feature	Default Setting
Area	Authentication type: 0 (no authentication). Default cost: 1. Range: Disabled. Stub: No stub area defined. NSSA: No NSSA area defined.
Auto cost	100 Mbps.
Default-information originate	Disabled. When enabled, the default metric setting is 10, and the external route type default is Type 2.
Default metric	Built-in, automatic metric translation, as appropriate for each routing protocol.
Distance OSPF	dist1 (all routes within an area): 110. dist2 (all routes from one area to another): 110. and dist3 (routes from other routing domains): 110.
OSPF database filter	Disabled. All outgoing link-state advertisements (LSAs) are flooded to the interface.
IP OSPF name lookup	Disabled.
Log adjacency changes	Enabled.
Neighbor	None specified.
Neighbor database filter	Disabled. All outgoing LSAs are flooded to the neighbor.
Network area	Disabled.
Router ID	No OSPF routing process defined.
Summary address	Disabled.
Timers LSA group pacing	240 seconds.
Timers shortest path first (spf)	spf delay: 5 seconds. spf-holdtime: 10 seconds.
Virtual link	No area ID or router ID defined. Hello interval: 10 seconds. Retransmit interval: 5 seconds. Transmit delay: 1 second. Dead interval: 40 seconds. Authentication key: no key predefined. Message-digest key (MD5): no key predefined.

Configuring Basic OSPF Parameters

Enabling OSPF requires that you create an OSPF routing process, specify the range of IP addresses to be associated with the routing process, and assign area IDs to be associated with that range.

Beginning in privileged EXEC mode, follow these steps to enable OSPF:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**router ospf** *process-id*	Enable OSPF routing, and enter router configuration mode. The process ID is an internally used identification parameter that is locally assigned and can be any positive integer. Each OSPF routing process has a unique value.
Step 3	**network** *address wildcard-mask* **area** *area-id*	Define an interface on which OSPF runs and the area ID for that interface. You can use the wildcard-mask to use a single command to define one or more multiple interfaces to be associated with a specific OSPF area. The area ID can be a decimal value or an IP address.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show ip protocols**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To terminate an OSPF routing process, use the **no router ospf** *process-id* global configuration command.

This example shows how to configure an OSPF routing process and assign it a process number of 109:

```
Switch(config)# router ospf 109
Switch(config-router)# network 131.108.0.0 255.255.255.0 area 24
```

Configuring OSPF Interfaces

You can use the **ip ospf** interface configuration commands to modify interface-specific OSPF parameters. You are not required to modify any of these parameters, but some interface parameters (hello interval, dead interval, and authentication key) must be consistent across all routers in an attached network. If you modify these parameters, be sure all routers in the network have compatible values.

Note The **ip ospf** interface configuration commands are all optional.

Beginning in privileged EXEC mode, follow these steps to modify OSPF interface parameters:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and specify the Layer 3 interface to configure.
Step 3	**no shutdown**	Enable the interface if necessary. By default, UNIs are disabled and NNIs are enabled.

	Command	Purpose
Step 4	**ip ospf** *cost*	(Optional) Explicitly specify the cost of sending a packet on the interface.
Step 5	**ip ospf retransmit-interval** *seconds*	(Optional) Specify the number of seconds between link state advertisement transmissions. The range is 1 to 65535 seconds. The default is 5 seconds.
Step 6	**ip ospf transmit-delay** *seconds*	(Optional) Set the estimated number of seconds to wait before sending a link state update packet. The range is 1 to 65535 seconds. The default is 1 second.
Step 7	**ip ospf priority** *number*	(Optional) Set priority to help find the OSPF designated router for a network. The range is from 0 to 255. The default is 1.
Step 8	**ip ospf hello-interval** *seconds*	(Optional) Set the number of seconds between hello packets sent on an OSPF interface. The value must be the same for all nodes on a network. The range is 1 to 65535 seconds. The default is 10 seconds.
Step 9	**ip ospf dead-interval** *seconds*	(Optional) Set the number of seconds after the last device hello packet was seen before its neighbors declare the OSPF router to be down. The value must be the same for all nodes on a network. The range is 1 to 65535 seconds. The default is 4 times the hello interval.
Step 10	**ip ospf authentication-key** *key*	(Optional) Assign a password to be used by neighboring OSPF routers. The password can be any string of keyboard-entered characters up to 8 bytes in length. All neighboring routers on the same network must have the same password to exchange OSPF information.
Step 11	**ip ospf message digest-key** *keyid* **md5** *key*	(Optional) Enable MDS authentication. • *keyid*—An identifier from 1 to 255. • *key*—An alphanumeric password of up to 16 bytes.
Step 12	**ip ospf database-filter all out**	(Optional) Block flooding of OSPF LSA packets to the interface. By default, OSPF floods new LSAs over all interfaces in the same area, except the interface on which the LSA arrives.
Step 13	**end**	Return to privileged EXEC mode.
Step 14	**show ip ospf interface** [*interface-name*]	Display OSPF-related interface information.
Step 15	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no** form of these commands to remove the configured parameter value or return to the default value.

Configuring OSPF Area Parameters

You can optionally configure several OSPF area parameters. These parameters include authentication for password-based protection against unauthorized access to an area, stub areas, and not-so-stubby-areas (NSSAs). *Stub areas* are areas into which information on external routes is not sent. Instead, the area border router (ABR) generates a default external route into the stub area for destinations outside the autonomous system (AS). An NSSA does not flood all LSAs from the core into the area, but can import AS external routes within the area by redistribution.

Route summarization is the consolidation of advertised addresses into a single summary route to be advertised by other areas. If network numbers are contiguous, you can use the **area range** router configuration command to configure the ABR to advertise a summary route that covers all networks in the range.

Note The OSPF **area** router configuration commands are all optional.

Beginning in privileged EXEC mode, follow these steps to configure area parameters:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**router ospf** *process-id*	Enable OSPF routing, and enter router configuration mode.
Step 3	**area** *area-id* **authentication**	(Optional) Allow password-based protection against unauthorized access to the identified area. The identifier can be either a decimal value or an IP address.
Step 4	**area** *area-id* **authentication message-digest**	(Optional) Enable MD5 authentication on the area.
Step 5	**area** *area-id* **stub** [**no-summary**]	(Optional) Define an area as a stub area. The **no-summary** keyword prevents an ABR from sending summary link advertisements into the stub area.
Step 6	**area** *area-id* **nssa** [**no-redistribution**] [**default-information-originate**] [**no-summary**]	(Optional) Defines an area as a not-so-stubby-area. Every router within the same area must agree that the area is NSSA. Select one of these keywords: • **no-redistribution**—Select when the router is an NSSA ABR and you want the **redistribute** command to import routes into normal areas, but not into the NSSA. • **default-information-originate**—Select on an ABR to allow importing type 7 LSAs into the NSSA. • **no-redistribution**—Select to not send summary LSAs into the NSSA.
Step 7	**area** *area-id* **range** *address mask*	(Optional) Specify an address range for which a single route is advertised. Use this command only with area border routers.
Step 8	**end**	Return to privileged EXEC mode.
Step 9	**show ip ospf** [*process-id*] **show ip ospf** [*process-id* [*area-id*]] **database**	Display information about the OSPF routing process in general or for a specific process ID to verify configuration. Display lists of information related to the OSPF database for a specific router.
Step 10	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no** form of these commands to remove the configured parameter value or to return to the default value.

Configuring Other OSPF Parameters

You can optionally configure other OSPF parameters in router configuration mode.

- Route summarization: When redistributing routes from other protocols as described in the "Using Route Maps to Redistribute Routing Information" section on page 32-73, each route is advertised individually in an external LSA. To help decrease the size of the OSPF link state database, you can use the **summary-address** router configuration command to advertise a single router for all the redistributed routes included in a specified network address and mask.
- Virtual links: In OSPF, all areas must be connected to a backbone area. You can establish a virtual link in case of a backbone-continuity break by configuring two Area Border Routers as endpoints of a virtual link. Configuration information includes the identity of the other virtual endpoint (the other ABR) and the nonbackbone link that the two routers have in common (the transit area). Virtual links cannot be configured through a stub area.
- Default route: When you specifically configure redistribution of routes into an OSPF routing domain, the route automatically becomes an autonomous system boundary router (ASBR). You can force the ASBR to generate a default route into the OSPF routing domain.
- Domain Name Server (DNS) names for use in all OSPF **show** privileged EXEC command displays makes it easier to identify a router than displaying it by router ID or neighbor ID.
- Default Metrics: OSPF calculates the OSPF metric for an interface according to the bandwidth of the interface. The metric is calculated as *ref-bw* divided by bandwidth, where *ref* is 10 by default, and bandwidth (*bw*) is specified by the **bandwidth** interface configuration command. For multiple links with high bandwidth, you can specify a larger number to differentiate the cost on those links.
- Administrative distance is a rating of the trustworthiness of a routing information source, an integer between 0 and 255, with a higher value meaning a lower trust rating. An administrative distance of 255 means the routing information source cannot be trusted at all and should be ignored. OSPF uses three different administrative distances: routes within an area (interarea), routes to another area (interarea), and routes from another routing domain learned through redistribution (external). You can change any of the distance values.
- Passive interfaces: Because interfaces between two devices on an Ethernet represent only one network segment, to prevent OSPF from sending hello packets for the sending interface, you must configure the sending device to be a passive interface. Both devices can identify each other through the hello packet for the receiving interface.
- Route calculation timers: You can configure the delay time between when OSPF receives a topology change and when it starts the shortest path first (SPF) calculation and the hold time between two SPF calculations.
- Log neighbor changes: You can configure the router to send a syslog message when an OSPF neighbor state changes, providing a high-level view of changes in the router.

Beginning in privileged EXEC mode, follow these steps to configure these OSPF parameters:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**router ospf** *process-id*	Enable OSPF routing, and enter router configuration mode.
Step 3	**summary-address** *address mask*	(Optional) Specify an address and IP subnet mask for redistributed routes so that only one summary route is advertised.

	Command	Purpose
Step 4	**area** *area-id* **virtual-link** router-id [**hello-interval** *seconds*] [**retransmit-interval** *seconds*] [**trans**] [[**authentication-key** *key*] \| **message-digest-key** *keyid* **md5** *key*]]	(Optional) Establish a virtual link and set its parameters. See the "Configuring OSPF Interfaces" section on page 32-26 for parameter definitions and Table 32-5 on page 32-24 for virtual link defaults.
Step 5	**default-information originate** [**always**] [**metric** *metric-value*] [**metric-type** *type-value*] [**route-map** *map-name*]	(Optional) Force the ASBR to generate a default route into the OSPF routing domain. Parameters are all optional.
Step 6	**ip ospf name-lookup**	(Optional) Configure DNS name lookup. The default is disabled.
Step 7	**ip auto-cost reference-bandwidth** *ref-bw*	(Optional) Specify an address range for which a single route will be advertised. Use this command only with area border routers.
Step 8	**distance ospf** {[**inter-area** *dist1*] [**inter-area** *dist2*] [**external** *dist3*]}	(Optional) Change the OSPF distance values. The default distance for each type of route is 110. The range is 1 to 255.
Step 9	**passive-interface** *type number*	(Optional) Suppress the sending of hello packets through the specified interface.
Step 10	**timers throttle spf** *spf-delay spf-holdtime spf-wait*	(Optional) Configure route calculation timers. • *spf-delay*—Delay between receiving a change to SPF calculation. The range is from 1 to 600000. miliseconds. • *spf-holdtime*—Delay between first and second SPF calculation. The range is form 1 to 600000 in milliseconds. • *spf-wait*—Maximum wait time in milliseconds for SPF calculations. The range is from 1 to 600000 in milliseconds.
Step 11	**ospf log-adj-changes**	(Optional) Send syslog message when a neighbor state changes.
Step 12	**end**	Return to privileged EXEC mode.
Step 13	**show ip ospf** [*process-id* [*area-id*]] **database**	Display lists of information related to the OSPF database for a specific router. For some of the keyword options, see the "Monitoring OSPF" section on page 32-31.
Step 14	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Changing LSA Group Pacing

The OSPF LSA group pacing feature allows the router to group OSPF LSAs and pace the refreshing, check-summing, and aging functions for more efficient router use. This feature is enabled by default with a 4-minute default pacing interval, and you will not usually need to modify this parameter. The optimum group pacing interval is inversely proportional to the number of LSAs the router is refreshing, check-summing, and aging. For example, if you have approximately 10,000 LSAs in the database, decreasing the pacing interval would benefit you. If you have a very small database (40 to 100 LSAs), increasing the pacing interval to 10 to 20 minutes might benefit you slightly.

Beginning in privileged EXEC mode, follow these steps to configure OSPF LSA pacing:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**router ospf** *process-id*	Enable OSPF routing, and enter router configuration mode.

	Command	Purpose
Step 3	**timers lsa-group-pacing** *seconds*	Change the group pacing of LSAs.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default value, use the **no timers lsa-group-pacing** router configuration command.

Configuring a Loopback Interface

OSPF uses the highest IP address configured on the interfaces as its router ID. If this interface is down or removed, the OSPF process must recalculate a new router ID and resend all its routing information out its interfaces. If a loopback interface is configured with an IP address, OSPF uses this IP address as its router ID, even if other interfaces have higher IP addresses. Because loopback interfaces never fail, this provides greater stability. OSPF automatically prefers a loopback interface over other interfaces, and it chooses the highest IP address among all loopback interfaces.

Beginning in privileged EXEC mode, follow these steps to configure a loopback interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface loopback 0**	Create a loopback interface, and enter interface configuration mode.
Step 3	**ip address** *address mask*	Assign an IP address to this interface.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show ip interface**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no interface loopback 0** global configuration command to disable the loopback interface.

Monitoring OSPF

You can display specific statistics such as the contents of IP routing tables, caches, and databases.

Table 32-6 lists some of the privileged EXEC commands for displaying statistics. For more **show ip ospf database** privileged EXEC command options and for explanations of fields in the resulting display, see the *Cisco IOS IP Command Reference, Volume 2 of 3: Routing Protocols, Release 12.2*.

Table 32-6 Show IP OSPF Statistics Commands

Command	Purpose
show ip ospf [*process-id*]	Display general information about OSPF routing processes.
show ip ospf [*process-id*] **database** [**router**] [*link-state-id*] **show ip ospf** [*process-id*] **database** [**router**] [**self-originate**] **show ip ospf** [*process-id*] **database** [**router**] [**adv-router** [*ip-address*]] **show ip ospf** [*process-id*] **database** [**network**] [*link-state-id*] **show ip ospf** [*process-id*] **database** [**summary**] [*link-state-id*] **show ip ospf** [*process-id*] **database** [**asbr-summary**] [*link-state-id*] **show ip ospf** [*process-id*] **database** [**external**] [*link-state-id*] **show ip ospf** [*process-id area-id*] **database** [**database-summary**]	Display lists of information related to the OSPF database.
show ip ospf border-routes	Display the internal OSPF routing ABR and ASBR table entries.
show ip ospf interface [*interface-name*]	Display OSPF-related interface information.
show ip ospf neighbor [*interface-name*] [*neighbor-id*] **detail**	Display OSPF interface neighbor information.
show ip ospf virtual-links	Display OSPF-related virtual links information.

Configuring EIGRP

Enhanced IGRP (EIGRP) is a Cisco proprietary enhanced version of the IGRP. EIGRP uses the same distance vector algorithm and distance information as IGRP; however, the convergence properties and the operating efficiency of EIGRP are significantly improved.

The convergence technology employs an algorithm referred to as the Diffusing Update Algorithm (DUAL), which guarantees loop-free operation at every instant throughout a route computation and allows all devices involved in a topology change to synchronize at the same time. Routers that are not affected by topology changes are not involved in recomputations.

IP EIGRP provides increased network width. With RIP, the largest possible width of your network is 15 hops. Because the EIGRP metric is large enough to support thousands of hops, the only barrier to expanding the network is the transport-layer hop counter. EIGRP increments the transport control field only when an IP packet has traversed 15 routers and the next hop to the destination was learned through EIGRP. When a RIP route is used as the next hop to the destination, the transport control field is incremented as usual.

EIGRP offers these features:

- Fast convergence.
- Incremental updates when the state of a destination changes, instead of sending the entire contents of the routing table, minimizing the bandwidth required for EIGRP packets.
- Less CPU usage because full update packets need not be processed each time they are received.
- Protocol-independent neighbor discovery mechanism to learn about neighboring routers.
- Variable-length subnet masks (VLSMs).

- Arbitrary route summarization.
- EIGRP scales to large networks.

EIGRP has these four basic components:

- *Neighbor discovery and recovery* is the process that routers use to dynamically learn of other routers on their directly attached networks. Routers must also discover when their neighbors become unreachable or inoperative. Neighbor discovery and recovery is achieved with low overhead by periodically sending small hello packets. As long as hello packets are received, the Cisco IOS software can learn that a neighbor is alive and functioning. When this status is determined, the neighboring routers can exchange routing information.
- *The reliable transport protocol* is responsible for guaranteed, ordered delivery of EIGRP packets to all neighbors. It supports intermixed transmission of multicast and unicast packets. Some EIGRP packets must be sent reliably, and others need not be. For efficiency, reliability is provided only when necessary. For example, on a multiaccess network that has multicast capabilities (such as Ethernet), it is not necessary to send hellos reliably to all neighbors individually. Therefore, EIGRP sends a single multicast hello with an indication in the packet informing the receivers that the packet need not be acknowledged. Other types of packets (such as updates) require acknowledgment, which is shown in the packet. The reliable transport has a provision to send multicast packets quickly when there are unacknowledged packets pending. Doing so helps ensure that convergence time remains low in the presence of varying speed links.
- *The DUAL finite state machine* embodies the decision process for all route computations. It tracks all routes advertised by all neighbors. DUAL uses the distance information (known as a metric) to select efficient, loop-free paths. DUAL selects routes to be inserted into a routing table based on feasible successors. A successor is a neighboring router used for packet forwarding that has a least-cost path to a destination that is guaranteed not to be part of a routing loop. When there are no feasible successors, but there are neighbors advertising the destination, a recomputation must occur. This is the process whereby a new successor is determined. The amount of time it takes to recompute the route affects the convergence time. Recomputation is processor-intensive; it is advantageous to avoid recomputation if it is not necessary. When a topology change occurs, DUAL tests for feasible successors. If there are feasible successors, it uses any it finds to avoid unnecessary recomputation.
- The *protocol-dependent modules* are responsible for network layer protocol-specific tasks. An example is the IP EIGRP module, which is responsible for sending and receiving EIGRP packets that are encapsulated in IP. It is also responsible for parsing EIGRP packets and informing DUAL of the new information received. EIGRP asks DUAL to make routing decisions, but the results are stored in the IP routing table. EIGRP is also responsible for redistributing routes learned by other IP routing protocols.

These sections contain this configuration information:

- Default EIGRP Configuration, page 32-34
- Configuring Basic EIGRP Parameters, page 32-35
- Configuring EIGRP Interfaces, page 32-36
- Configuring EIGRP Route Authentication, page 32-37
- Monitoring and Maintaining EIGRP, page 32-38

Default EIGRP Configuration

Table 32-7, Part 1 shows the default EIGRP configuration.

Table 32-7, Part 1 Default EIGRP Configuration

Feature	Default Setting
Auto summary	Enabled. Subprefixes are summarized to the classful network boundary when crossing classful network boundaries.
Default-information	Exterior routes are accepted and default information is passed between EIGRP processes when doing redistribution.
Default metric	Only connected routes and interface static routes can be redistributed without a default metric. The metric includes: • Bandwidth: 0 or greater kbps. • Delay (tens of microseconds): 0 or any positive number that is a multiple of 39.1 nanoseconds. • Reliability: any number between 0 and 255 (255 means 100 percent reliability). • Loading: effective bandwidth as a number between 0 and 255 (255 is 100 percent loading). • MTU: maximum transmission unit size of the route in bytes. 0 or any positive integer.
Distance	Internal distance: 90. External distance: 170.
EIGRP log-neighbor changes	Disabled. No adjacency changes logged.
IP authentication key-chain	No authentication provided.
IP authentication mode	No authentication provided.
IP bandwidth-percent	50 percent.
IP hello interval	For low-speed nonbroadcast multiaccess (NBMA) networks: 60 seconds; all other networks: 5 seconds.
IP hold-time	For low-speed NBMA networks: 180 seconds; all other networks: 15 seconds.
IP split-horizon	Enabled.
IP summary address	No summary aggregate addresses are predefined.
Metric weights	tos: 0; k1 and k3: 1; k2, k4, and k5: 0
Network	None specified.
Offset-list	Disabled.
Router EIGRP	Disabled.
Set metric	No metric set in the route map.
Traffic-share	Distributed proportionately to the ratios of the metrics.
Variance	1 (equal-cost load balancing).

To create an EIGRP routing process, you must enable EIGRP and associate networks. EIGRP sends updates to the interfaces in the specified networks. If you do not specify an interface network, it is not advertised in any EIGRP update.

Configuring Basic EIGRP Parameters

Beginning in privileged EXEC mode, follow these steps to configure EIGRP. Configuring the routing process is required; other steps are optional:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**router eigrp** *autonomous-system*	Enable an EIGRP routing process, and enter router configuration mode. The AS number identifies the routes to other EIGRP routers and is used to tag routing information.
Step 3	**network** *network-number*	Associate networks with an EIGRP routing process. EIGRP sends updates to the interfaces in the specified networks.
Step 4	**eigrp log-neighbor-changes**	(Optional) Enable logging of EIGRP neighbor changes to monitor routing system stability.
Step 5	**metric weights** *tos k1 k2 k3 k4 k5*	(Optional) Adjust the EIGRP metric. Although the defaults have been carefully set to provide excellent operation in most networks, you can adjust them. **Caution** Setting metrics is complex and is not recommended without guidance from an experienced network designer.
Step 6	**offset list** [*access-list number* \| *name*] {**in** \| **out**} *offset* [*type number*]	(Optional) Apply an offset list to routing metrics to increase incoming and outgoing metrics to routes learned through EIGRP. You can limit the offset list with an access list or an interface.
Step 7	**no auto-summary**	(Optional) Disable automatic summarization of subnet routes into network-level routes.
Step 8	**ip summary-address eigrp** *autonomous-system-number address mask*	(Optional) Configure a summary aggregate.
Step 9	**end**	Return to privileged EXEC mode.
Step 10	**show ip protocols**	Verify your entries.
Step 11	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no** forms of these commands to disable the feature or return the setting to the default value.

Configuring EIGRP Interfaces

Other optional EIGRP parameters can be configured on an interface basis.

Beginning in privileged EXEC mode, follow these steps to configure EIGRP interfaces:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and specify the Layer 3 interface to configure.
Step 3	**no shutdown**	Enable the interface if necessary. By default, UNIs are disabled and NNIs are enabled.
Step 4	**ip bandwidth-percent eigrp** *percent*	(Optional) Configure the percentage of bandwidth that can be used by EIGRP on an interface. The default is 50 percent.
Step 5	**ip summary-address eigrp** *autonomous-system-number address mask*	(Optional) Configure a summary aggregate address for a specified interface (not usually necessary if auto-summary is enabled).
Step 6	**ip hello-interval eigrp** *autonomous-system-number seconds*	(Optional) Change the hello time interval for an EIGRP routing process. The range is 1 to 65535 seconds. The default is 60 seconds for low-speed NBMA networks and 5 seconds for all other networks.
Step 7	**ip hold-time eigrp** *autonomous-system-number seconds*	(Optional) Change the hold time interval for an EIGRP routing process. The range is 1 to 65535 seconds. The default is 180 seconds for low-speed NBMA networks and 15 seconds for all other networks. **Caution** Do not adjust the hold time without consulting Cisco technical support.
Step 8	**no ip split-horizon eigrp** *autonomous-system-number*	(Optional) Disable split horizon to allow route information to be advertised by a router out any interface from which that information originated.
Step 9	**end**	Return to privileged EXEC mode.
Step 10	**show ip eigrp interface**	Display which interfaces EIGRP is active on and information about EIGRP relating to those interfaces.
Step 11	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no** forms of these commands to disable the feature or return the setting to the default value.

Configuring EIGRP Route Authentication

EIGRP route authentication provides MD5 authentication of routing updates from the EIGRP routing protocol to prevent the introduction of unauthorized or false routing messages from unapproved sources.

Beginning in privileged EXEC mode, follow these steps to enable authentication:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and specify the Layer 3 interface to configure.
Step 3	**no shutdown**	Enable the interface if necessary. By default, UNIs are disabled and NNIs are enabled.
Step 4	**ip authentication mode eigrp** *autonomous-system* **md5**	Enable MD5 authentication in IP EIGRP packets.
Step 5	**ip authentication key-chain eigrp** *autonomous-system key-chain*	Enable authentication of IP EIGRP packets.
Step 6	**exit**	Return to global configuration mode.
Step 7	**key chain** *name-of-chain*	Identify a key chain and enter key-chain configuration mode. Match the name configured in Step 4.
Step 8	**key** *number*	In key-chain configuration mode, identify the key number.
Step 9	**key-string** *text*	In key-chain key configuration mode, identify the key string.
Step 10	**accept-lifetime** *start-time* {**infinite** \| *end-time* \| **duration** *seconds*}	(Optional) Specify the time period during which the key can be received. The *start-time* and *end-time* syntax can be either *hh*:*mm*:*ss Month date year* or *hh*:*mm*:*ss date Month year*. The default is forever with the default *start-time* and the earliest acceptable date as January 1, 1993. The default *end-time* and **duration** is **infinite**.
Step 11	**send-lifetime** *start-time* {**infinite** \| *end-time* \| **duration** *seconds*}	(Optional) Specify the time period during which the key can be sent. The *start-time* and *end-time* syntax can be either *hh*:*mm*:*ss Month date year* or *hh*:*mm*:*ss date Month year*. The default is forever with the default *start-time* and the earliest acceptable date as January 1, 1993. The default *end-time* and **duration** is **infinite**.
Step 12	**end**	Return to privileged EXEC mode.
Step 13	**show key chain**	Display authentication key information.
Step 14	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no** forms of these commands to disable the feature or to return the setting to the default value.

Monitoring and Maintaining EIGRP

You can delete neighbors from the neighbor table. You can also display various EIGRP routing statistics. Table 32-8 lists the privileged EXEC commands for deleting neighbors and displaying statistics. For explanations of fields in the resulting display, see the *Cisco IOS IP Command Reference, Volume 2 of 3: Routing Protocols, Release 12.2*.

Table 32-8 IP EIGRP Clear and Show Commands

Command	Purpose
clear ip eigrp neighbors [*if-address* \| *interface*]	Delete neighbors from the neighbor table.
show ip eigrp interface [*interface*] [*as number*]	Display information about interfaces configured for EIGRP.
show ip eigrp neighbors [*type-number*]	Display EIGRP discovered neighbors.
show ip eigrp topology [*autonomous-system-number*] \| [[*ip-address*] *mask*]]	Display the EIGRP topology table for a given process.
show ip eigrp traffic [*autonomous-system-number*]	Display the number of packets sent and received for all or a specified EIGRP process.

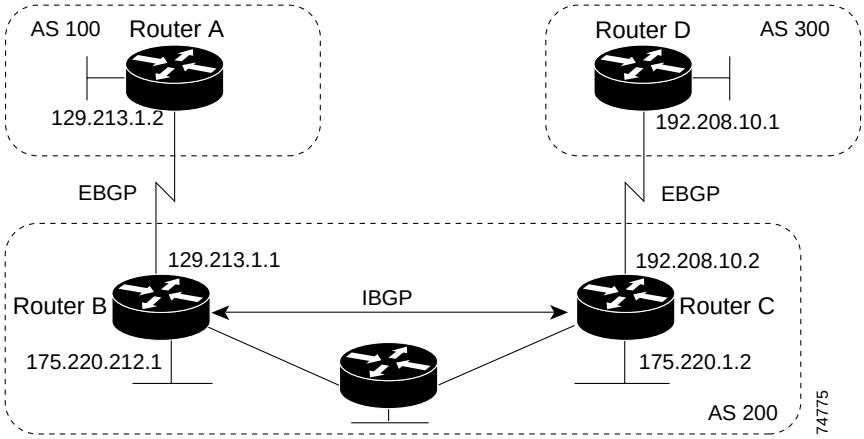
Configuring BGP

The Border Gateway Protocol (BGP) is an exterior gateway protocol used to set up an interdomain routing system that guarantees the loop-free exchange of routing information between autonomous systems. Autonomous systems are made up of routers that operate under the same administration and that run Interior Gateway Protocols (IGPs), such as RIP or OSPF, within their boundaries and that interconnect by using an Exterior Gateway Protocol (EGP). BGP Version 4 is the standard EGP for interdomain routing in the Internet. The protocol is defined in RFCs 1163, 1267, and 1771. You can find detailed information about BGP in *Internet Routing Architectures*, published by Cisco Press, and in the "Configuring BGP" chapter in the *Cisco IOS IP and IP Routing Configuration Guide*.

For details about BGP commands and keywords, see the "IP Routing Protocols" part of the *Cisco IOS IP Command Reference, Volume 2 of 3: Routing Protocols, Release 12.2*. For a list of BGP commands that are visible but not supported by the switch, see Appendix C, "Unsupported Commands in Cisco IOS Release 12.2(25)EX."

Routers that belong to the same autonomous system (AS) and that exchange BGP updates run *internal BGP* (IBGP), and routers that belong to different autonomous systems and that exchange BGP updates run *external BGP* (EBGP). Most configuration commands are the same for configuring EBGP and IBGP. The difference is that the routing updates are exchanged either between autonomous systems (EBGP) or within an AS (IBGP). Figure 32-4 shows a network that is running both EBGP and IBGP.

Figure 32-4 EBGP, IBGP, and Multiple Autonomous Systems



Before exchanging information with an external AS, BGP ensures that networks within the AS can be reached by defining internal BGP peering among routers within the AS and by redistributing BGP routing information to IGPs that run within the AS, such as IGRP and OSPF.

Routers that run a BGP routing process are often referred to as BGP *speakers*. BGP uses the Transmission Control Protocol (TCP) as its transport protocol (specifically port 179). Two BGP speakers that have a TCP connection to each other for exchanging routing information are known as *peers* or *neighbors*. In Figure 32-4, Routers A and B are BGP peers, as are Routers B and C and Routers C and D. The routing information is a series of AS numbers that describe the full path to the destination network. BGP uses this information to construct a loop-free map of autonomous systems.

The network has these characteristics:

- Routers A and B are running EBGP, and Routers B and C are running IBGP. Note that the EBGP peers are directly connected and that the IBGP peers are not. As long as there is an IGP running that allows the two neighbors to reach one another, IBGP peers do not have to be directly connected.
- All BGP speakers within an AS must establish a peer relationship with each other. That is, the BGP speakers within an AS must be fully meshed logically. BGP4 provides two techniques that reduce the requirement for a logical full mesh: *confederations* and *route reflectors*.
- AS 200 is a *transit AS* for AS 100 and AS 300—that is, AS 200 is used to transfer packets between AS 100 and AS 300.

BGP peers initially exchange their full BGP routing tables and then send only incremental updates. BGP peers also exchange keepalive messages (to ensure that the connection is up) and notification messages (in response to errors or special conditions).

In BGP, each route consists of a network number, a list of autonomous systems that information has passed through (the *autonomous system path*), and a list of other *path attributes*. The primary function of a BGP system is to exchange network reachability information, including information about the list of AS paths, with other BGP systems. This information can be used to determine AS connectivity, to prune routing loops, and to enforce AS-level policy decisions.

A router or switch running Cisco IOS does not select or use an IBGP route unless it has a route available to the next-hop router and it has received synchronization from an IGP (unless IGP synchronization is disabled). When multiple routes are available, BGP bases its path selection on *attribute* values. See the "Configuring BGP Decision Attributes" section on page 32-46 for information about BGP attributes.

BGP Version 4 supports classless interdomain routing (CIDR) so you can reduce the size of your routing tables by creating aggregate routes, resulting in *supernets*. CIDR eliminates the concept of network classes within BGP and supports the advertising of IP prefixes.

These sections contain this configuration information:

- Default BGP Configuration, page 32-40
- Enabling BGP Routing, page 32-42
- Managing Routing Policy Changes, page 32-45
- Configuring BGP Decision Attributes, page 32-46
- Configuring BGP Filtering with Route Maps, page 32-48
- Configuring BGP Filtering by Neighbor, page 32-49
- Configuring Prefix Lists for BGP Filtering, page 32-50
- Configuring BGP Community Filtering, page 32-51
- Configuring BGP Neighbors and Peer Groups, page 32-52
- Configuring Aggregate Addresses, page 32-54
- Configuring Routing Domain Confederations, page 32-55
- Configuring BGP Route Reflectors, page 32-56
- Configuring Route Dampening, page 32-57
- Monitoring and Maintaining BGP, page 32-58

For detailed descriptions of BGP configuration, see the "Configuring BGP" chapter in the "IP Routing Protocols" part of the Cisco IOS IP Configuration Guide, Release 12.2. For details about specific commands, see the *Cisco IOS IP Command Reference, Volume 2 of 3: Routing Protocols, Release 12.2*.

For a list of BGP commands that are visible but not supported by the switch, see Appendix C, "Unsupported Commands in Cisco IOS Release 12.2(25)EX."

Default BGP Configuration

Table 32-9 shows the basic default BGP configuration. For the defaults for all characteristics, see the specific commands in the *Cisco IOS IP Command Reference, Volume 2 of 3: Routing Protocols, Release 12.2*.

Table 32-9 Default BGP Configuration

Feature	Default Setting
Aggregate address	Disabled: None defined.
AS path access list	None defined.
Auto summary	Enabled.
Best path	• The router considers *as-path* in choosing a route and does not compare similar routes from external BGP peers. • Compare router ID: Disabled.
BGP community list	• Number: None defined. When you permit a value for the community number, the list defaults to an implicit deny for everything else that has not been permitted. • Format: Cisco default format (32-bit number).

Table 32-9 Default BGP Configuration (continued)

Feature	Default Setting
BGP confederation identifier/peers	• Identifier: None configured. • Peers: None identified.
BGP Fast external fallover	Enabled.
BGP local preference	100. The range is 0 to 4294967295 with the higher value preferred.
BGP network	None specified; no backdoor route advertised.
BGP route dampening	Disabled by default. When enabled: • Half-life is 15 minutes. • Re-use is 750 (10-second increments). • Suppress is 2000 (10-second increments). • Max-suppress-time is 4 times half-life; 60 minutes.
BGP router ID	The IP address of a loopback interface if one is configured or the highest IP address configured for a physical interface on the router.
Default information originate (protocol or network redistribution)	Disabled.
Default metric	Built-in, automatic metric translations.
Distance	• External route administrative distance: 20 (acceptable values are from 1 to 255). • Internal route administrative distance: 200 (acceptable values are from 1 to 255). • Local route administrative distance: 200 (acceptable values are from 1 to 255).
Distribute list	• In (filter networks received in updates): Disabled. • Out (suppress networks from being advertised in updates): Disabled.
Internal route redistribution	Disabled.
IP prefix list	None defined.
Multi exit discriminator (MED)	• Always compare: Disabled. Does not compare MEDs for paths from neighbors in different autonomous systems. • Best path compare: Disabled. • MED missing as worst path: Disabled. • Deterministic MED comparison is disabled.

Table 32-9 Default BGP Configuration (continued)

Feature	Default Setting
Neighbor	• Advertisement interval: 30 seconds for external peers; 5 seconds for internal peers. • Change logging: Enabled. • Conditional advertisement: Disabled. • Default originate: No default route is sent to the neighbor. • Description: None. • Distribute list: None defined. • External BGP multihop: Only directly connected neighbors are allowed. • Filter list: None used. • Maximum number of prefixes received: No limit. • Next hop (router as next hop for BGP neighbor): Disabled. • Password: Disabled. • Peer group: None defined; no members assigned. • Prefix list: None specified. • Remote AS (add entry to neighbor BGP table): No peers defined. • Private AS number removal: Disabled. • Route maps: None applied to a peer. • Send community attributes: None sent to neighbors. • Shutdown or soft reconfiguration: Not enabled. • Timers: keepalive: 60 seconds; holdtime: 180 seconds. • Update source: Best local address. • Version: BGP Version 4. • Weight: Routes learned through BGP peer: 0; routes sourced by the local router: 32768.
Route reflector	None configured.
Synchronization (BGP and IGP)	Enabled.
Table map update	Disabled.
Timers	Keepalive: 60 seconds; holdtime: 180 seconds.

Enabling BGP Routing

To enable BGP routing, you establish a BGP routing process and define the local network. Because BGP must completely recognize the relationships with its neighbors, you must also specify a BGP neighbor.

BGP supports two kinds of neighbors: internal and external. *Internal neighbors* are in the same AS; *external neighbors* are in different autonomous systems. External neighbors are usually adjacent to each other and share a subnet, but internal neighbors can be anywhere in the same AS.

The switch supports the use of private AS numbers, usually assigned by service providers and given to systems whose routes are not advertised to external neighbors. The private AS numbers are from 64512 to 65535. You can configure external neighbors to remove private AS numbers from the AS path by using the **neighbor remove-private-as** router configuration command. Then when an update is passed to an external neighbor, if the AS path includes private AS numbers, these numbers are dropped.

If your AS will be passing traffic through it from another AS to a third AS, it is important to be consistent about the routes it advertises. If BGP advertised a route before all routers in the network had learned about the route through the IGP, the AS might receive traffic that some routers could not yet route. To prevent this from happening, BGP must wait until the IGP has propagated information across the AS so that BGP is *synchronized* with the IGP. Synchronization is enabled by default. If your AS does not pass traffic from one AS to another AS, or if all routers in your autonomous systems are running BGP, you can disable synchronization, which allows your network to carry fewer routes in the IGP and allows BGP to converge more quickly.

Beginning in privileged EXEC mode, follow these steps to enable BGP routing, establish a BGP routing process, and specify a neighbor:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip routing**	Enable IP routing (required only if IP routing is disabled).
Step 3	**router bgp** *autonomous-system*	Enable a BGP routing process, assign it an AS number, and enter router configuration mode. The AS number can be from 1 to 65535, with 64512 to 65535 designated as private autonomous numbers.
Step 4	**network** *network-number* [**mask** *network-mask*] [**route-map** *route-map-name*]	Configure a network as local to this AS, and enter it in the BGP table.
Step 5	**neighbor** {*ip-address* \| *peer-group-name*} **remote-as** *number*	Add an entry to the BGP neighbor table specifying that the neighbor identified by the IP address belongs to the specified AS. For EBGP, neighbors are usually directly connected, and the IP address is the address of the interface at the other end of the connection. For IBGP, the IP address can be the address of any of the router interfaces.
Step 6	**neighbor** {*ip-address* \| *peer-group-name*} **remove-private-as**	(Optional) Remove private AS numbers from the AS-path in outbound routing updates.
Step 7	**no synchronization**	(Optional) Disable synchronization between BGP and an IGP.
Step 8	**no auto-summary**	(Optional) Disable automatic network summarization. By default, when a subnet is redistributed from an IGP into BGP, only the network route is inserted into the BGP table.
Step 9	**bgp fast-external-fallover**	(Optional) Automatically reset a BGP session when a link between external neighbors goes down. By default, the session is not immediately reset.
Step 10	**end**	Return to privileged EXEC mode.
Step 11	**show ip bgp network** *network-number* **show ip bgp neighbor**	Verify the configuration.
Step 12	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no router bgp** *autonomous-system* global configuration command to remove a BGP AS. Use the **no network** *network-number* router configuration command to remove the network from the BGP table. Use the **no neighbor** {*ip-address* | *peer-group-name*} **remote-as** *number* router configuration command to remove a neighbor. Use the **no neighbor** {*ip-address* | *peer-group-name*} **remove-private-as** router configuration command to include private AS numbers in updates to a neighbor. Use the **synchronization** router configuration command to re-enable synchronization.

These examples show how to configure BGP on the routers in Figure 32-4.

Router A:

```
Switch(config)# router bgp 100
Switch(config-router)# neighbor 129.213.1.1 remote-as 200
```

Router B:

```
Switch(config)# router bgp 200
Switch(config-router)# neighbor 129.213.1.2 remote-as 100
Switch(config-router)# neighbor 175.220.1.2 remote-as 200
```

Router C:

```
Switch(config)# router bgp 200
Switch(config-router)# neighbor 175.220.212.1 remote-as 200
Switch(config-router)# neighbor 192.208.10.1 remote-as 300
```

Router D:

```
Switch(config)# router bgp 300
Switch(config-router)# neighbor 192.208.10.2 remote-as 200
```

To verify that BGP peers are running, use the **show ip bgp neighbors** privileged EXEC command. This is the output of this command on Router A:

```
Switch# show ip bgp neighbors

BGP neighbor is 129.213.1.1, remote AS 200, external link
 BGP version 4, remote router ID 175.220.212.1
 BGP state = established, table version = 3, up for 0:10:59
 Last read 0:00:29, hold time is 180, keepalive interval is 60 seconds
 Minimum time between advertisement runs is 30 seconds
 Received 2828 messages, 0 notifications, 0 in queue
 Sent 2826 messages, 0 notifications, 0 in queue
 Connections established 11; dropped 10
```

Anything other than *state = established* means that the peers are not running. The remote router ID is the highest IP address on that router (or the highest loopback interface). Each time the table is updated with new information, the table version number increments. A table version number that continually increments means that a route is flapping, causing continual routing updates.

For exterior protocols, a reference to an IP network from the **network** router configuration command controls only which networks are advertised. This is in contrast to Interior Gateway Protocols (IGPs), such as EIGRP, which also use the **network** command to specify where to send updates.

For detailed descriptions of BGP configuration, see the "IP Routing Protocols" part of the *Cisco IOS IP Configuration Guide, Release 12.2*. For details about specific commands, see the *Cisco IOS IP Command Reference, Volume 2 of 3: Routing Protocols, Release 12.2*. See Appendix C, "Unsupported Commands in Cisco IOS Release 12.2(25)EX," for a list of BGP commands that are visible but not supported by the switch.

Managing Routing Policy Changes

Routing policies for a peer include all the configurations that might affect inbound or outbound routing table updates. When you have defined two routers as BGP neighbors, they form a BGP connection and exchange routing information. If you later change a BGP filter, weight, distance, version, or timer, or make a similar configuration change, you must reset the BGP sessions so that the configuration changes take effect.

There are two types of reset, hard reset and soft reset. Cisco IOS Releases 12.1 and later support a soft reset without any prior configuration. To use a soft reset without preconfiguration, both BGP peers must support the soft route refresh capability, which is advertised in the OPEN message sent when the peers establish a TCP session. A soft reset allows the dynamic exchange of route refresh requests and routing information between BGP routers and the subsequent re-advertisement of the respective outbound routing table.

- When soft reset generates inbound updates from a neighbor, it is called *dynamic inbound soft reset*.
- When soft reset sends a set of updates to a neighbor, it is called *outbound soft reset*.

A soft inbound reset causes the new inbound policy to take effect. A soft outbound reset causes the new local outbound policy to take effect without resetting the BGP session. As a new set of updates is sent during outbound policy reset, a new inbound policy can also take effect.

Table 32-10 lists the advantages and disadvantages hard reset and soft reset.

Table 32-10 Advantages and Disadvantages of Hard and Soft Resets

Type of Reset	Advantages	Disadvantages
Hard reset	No memory overhead	The prefixes in the BGP, IP, and FIB tables provided by the neighbor are lost. Not recommended.
Outbound soft reset	No configuration, no storing of routing table updates	Does not reset inbound routing table updates.
Dynamic inbound soft reset	Does not clear the BGP session and cache Does not require storing of routing table updates and has no memory overhead	Both BGP routers must support the route refresh capability (in Cisco IOS Release 12.1 and later).

Beginning in privileged EXEC mode, follow these steps to learn if a BGP peer supports the route refresh capability and to reset the BGP session:

	Command	Purpose
Step 1	**show ip bgp neighbors**	Display whether a neighbor supports the route refresh capability. When supported, this message appears for the router: *Received route refresh capability from peer.*
Step 2	**clear ip bgp** {***** \| *address* \| *peer-group-name*}	Reset the routing table on the specified connection. • Enter an asterisk (*) to specify that all connections be reset. • Enter an IP *address* to specify the connection to be reset. • Enter a peer group name to reset the peer group.

	Command	Purpose
Step 3	**clear ip bgp** {***** \| *address* \| *peer-group-name*} **soft out**	(Optional) Perform an outbound soft reset to reset the inbound routing table on the specified connection. Use this command if route refresh is supported. • Enter an asterisk (*) to specify that all connections be reset. • Enter an IP *address* to specify the connection to be reset. • Enter a peer group name to reset the peer group.
Step 4	**show ip bgp** **show ip bgp neighbors**	Verify the reset by checking information about the routing table and about BGP neighbors.

Configuring BGP Decision Attributes

When a BGP speaker receives updates from multiple autonomous systems that describe different paths to the same destination, it must choose the single best path for reaching that destination. When chosen, the selected path is entered into the BGP routing table and propagated to its neighbors. The decision is based on the value of attributes that the update contains and other BGP-configurable factors.

When a BGP peer learns two EBGP paths for a prefix from a neighboring AS, it chooses the best path and inserts that path in the IP routing table. If BGP multipath support is enabled and the EBGP paths are learned from the same neighboring autonomous systems, instead of a single best path, multiple paths are installed in the IP routing table. Then, during packet switching, per-packet or per-destination load balancing is performed among the multiple paths. The **maximum-paths** router configuration command controls the number of paths allowed.

These factors summarize the order in which BGP evaluates the attributes for choosing the best path:

1. If the path specifies a next hop that is inaccessible, drop the update. The BGP next-hop attribute, automatically determined by the software, is the IP address of the next hop that is going to be used to reach a destination. For EBGP, this is usually the IP address of the neighbor specified by the **neighbor remote-as** router configuration command. You can disable next-hop processing by using route maps or the **neighbor next-hop-self** router configuration command.
2. Prefer the path with the largest weight (a Cisco proprietary parameter). The weight attribute is local to the router and not propagated in routing updates. By default, the weight attribute is 32768 for paths that the router originates and zero for other paths. Routes with the largest weight are preferred. You can use access lists, route maps, or the **neighbor weight** router configuration command to set weights.
3. Prefer the route with the highest local preference. Local preference is part of the routing update and exchanged among routers in the same AS. The default value of the local preference attribute is 100. You can set local preference by using the **bgp default local-preference** router configuration command or by using a route map.
4. Prefer the route that was originated by BGP running on the local router.
5. Prefer the route with the shortest AS path.
6. Prefer the route with the lowest origin type. An interior route or IGP is lower than a route learned by EGP, and an EGP-learned route is lower than one of unknown origin or learned in another way.
7. Prefer the route with the lowest multi -exit discriminator (MED) metric attribute if the neighboring AS is the same for all routes considered. You can configure the MED by using route maps or by using the **default-metric** router configuration command. When an update is sent to an IBGP peer, the MED is included.
8. Prefer the external (EBGP) path over the internal (IBGP) path.

9. Prefer the route that can be reached through the closest IGP neighbor (the lowest IGP metric). This means that the router will prefer the shortest internal path within the AS to reach the destination (the shortest path to the BGP next-hop).
10. If the following conditions are all true, insert the route for this path into the IP routing table:
 - Both the best route and this route are external.
 - Both the best route and this route are from the same neighboring autonomous system.
 - maximum-paths is enabled.
11. If multipath is not enabled, prefer the route with the lowest IP address value for the BGP router ID. The router ID is usually the highest IP address on the router or the loopback (virtual) address, but might be implementation-specific.

Beginning in privileged EXEC mode, follow these steps to configure some decision attributes:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**router bgp** *autonomous-system*	Enable a BGP routing process, assign it an AS number, and enter router configuration mode.
Step 3	**bgp best-path as-path ignore**	(Optional) Configure the router to ignore AS path length in selecting a route.
Step 4	**neighbor** {*ip-address* \| *peer-group-name*} **next-hop-self**	(Optional) Disable next-hop processing on BGP updates to a neighbor by entering a specific IP address to be used instead of the next-hop address.
Step 5	**neighbor** {*ip-address* \| *peer-group-name*} **weight** *weight*	(Optional) Assign a weight to a neighbor connection. Acceptable values are from 0 to 65535; the largest weight is the preferred route. Routes learned through another BGP peer have a default weight of 0; routes sourced by the local router have a default weight of 32768.
Step 6	**default-metric** *number*	(Optional) Set a MED metric to set preferred paths to external neighbors. All routes without a MED will also be set to this value. The range is 1 to 4294967295. The lowest value is the most desirable.
Step 7	**bgp bestpath med missing-as-worst**	(Optional) Configure the switch to consider a missing MED as having a value of infinity, making the path without a MED value the least desirable path.
Step 8	**bgp always-compare med**	(Optional) Configure the switch to compare MEDs for paths from neighbors in different autonomous systems. By default, MED comparison is only done among paths in the same AS.
Step 9	**bgp bestpath med confed**	(Optional) Configure the switch to consider the MED in choosing a path from among those advertised by different subautonomous systems within a confederation.
Step 10	**bgp deterministic med**	(Optional) Configure the switch to consider the MED variable when choosing among routes advertised by different peers in the same AS.

	Command	Purpose
Step 11	**bgp default local-preference** *value*	(Optional) Change the default local preference value. The range is 0 to 4294967295; the default value is 100. The highest local preference value is preferred.
Step 12	**maximum-paths** *number*	(Optional) Configure the number of paths to be added to the IP routing table. The default is to only enter the best path in the routing table. The range is from 1 to 8. Having multiple paths allows load balancing among the paths.
Step 13	**end**	Return to privileged EXEC mode.
Step 14	**show ip bgp** **show ip bgp neighbors**	Verify the reset by checking information about the routing table and about BGP neighbors.
Step 15	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no** form of each command to return to the default state.

Configuring BGP Filtering with Route Maps

Within BGP, route maps can be used to control and to modify routing information and to define the conditions by which routes are redistributed between routing domains. See the "Using Route Maps to Redistribute Routing Information" section on page 32-73 for more information about route maps. Each route map has a name that identifies the route map (*map tag*) and an optional sequence number.

Beginning in privileged EXEC mode, follow these steps to use a route map to disable next-hop processing:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**route-map** *map-tag* [[**permit** \| **deny**] \| *sequence-number*]]	Create a route map, and enter route-map configuration mode.
Step 3	**set ip next-hop** *ip-address* [*...ip-address*] [**peer-address**]	(Optional) Set a route map to disable next-hop processing • In an inbound route map, set the next hop of matching routes to be the neighbor peering address, overriding third-party next hops. • In an outbound route map of a BGP peer, set the next hop to the peering address of the local router, disabling the next-hop calculation.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show route-map** [*map-name*]	Display all route maps configured or only the one specified to verify configuration.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no route-map** *map-tag* command to delete the route map. Use the **no set ip next-hop** *ip-address* command to re-enable next-hop processing.

Configuring BGP Filtering by Neighbor

You can filter BGP advertisements by using AS-path filters, such as the **as-path access-list** global configuration command and the **neighbor filter-list** router configuration command. You can also use access lists with the **neighbor distribute-list** router configuration command. Distribute-list filters are applied to network numbers. See the "Controlling Advertising and Processing in Routing Updates" section on page 32-80 for information about the **distribute-list** command.

You can use route maps on a per-neighbor basis to filter updates and to modify various attributes. A route map can be applied to either inbound or outbound updates. Only the routes that pass the route map are sent or accepted in updates. On both inbound and outbound updates, matching is supported based on AS path, community, and network numbers. Autonomous system path matching requires the **match as-path access-lis**t route-map command, community based matching requires the **match community-list** route-map command, and network-based matching requires the **ip access-list** global configuration command.

Beginning in privileged EXEC mode, follow these steps to apply a per-neighbor route map:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**router bgp** *autonomous-system*	Enable a BGP routing process, assign it an AS number, and enter router configuration mode.
Step 3	**neighbor** {*ip-address* \| *peer-group name*} **distribute-list** {*access-list-number* \| *name*} {**in** \| **out**}	(Optional) Filter BGP routing updates to or from neighbors as specified in an access list. **Note** You can also use the **neighbor prefix-list** router configuration command to filter updates, but you cannot use both commands to configure the same BGP peer.
Step 4	**neighbor** {*ip-address* \| *peer-group name*} **route-map** *map-tag* {**in** \| **out**}	(Optional) Apply a route map to filter an incoming or outgoing route.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show ip bgp neighbors**	Verify the configuration.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no neighbor distribute-list** command to remove the access list from the neighbor. Use the **no neighbor route-map** *map-tag* router configuration command to remove the route map from the neighbor.

Another method of filtering is to specify an access list filter on both incoming and outbound updates, based on the BGP autonomous system paths. Each filter is an access list based on regular expressions. (See the "Regular Expressions" appendix in the *Cisco IOS Dial Technologies Command Reference, Release 12.2* for more information on forming regular expressions.) To use this method, define an autonomous system path access list, and apply it to updates to and from particular neighbors.

Beginning in privileged EXEC mode, follow these steps to configure BGP path filtering:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip as-path access-list** *access-list-number* {**permit** \| **deny**} *as-regular-expressions*	Define a BGP-related access list.

	Command	Purpose
Step 3	**router bgp** *autonomous-system*	Enter BGP router configuration mode.
Step 4	**neighbor** {*ip-address* \| *peer-group name*} **filter-list** {*access-list-number* \| *name*} {**in** \| **out** \| **weight** *weight*}	Establish a BGP filter based on an access list.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show ip bgp neighbors** [**paths** regular-expression]	Verify the configuration.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Configuring Prefix Lists for BGP Filtering

You can use prefix lists as an alternative to access lists in many BGP route filtering commands, including the **neighbor distribute-list** router configuration command. The advantages of using prefix lists include performance improvements in loading and lookup of large lists, incremental update support, easier CLI configuration, and greater flexibility.

Filtering by a prefix list involves matching the prefixes of routes with those listed in the prefix list, as when matching access lists. When there is a match, the route is used. Whether a prefix is permitted or denied is based upon these rules:

- An empty prefix list permits all prefixes.
- An implicit deny is assumed if a given prefix does not match any entries in a prefix list.
- When multiple entries of a prefix list match a given prefix, the sequence number of a prefix list entry identifies the entry with the lowest sequence number.

By default, sequence numbers are generated automatically and incremented in units of five. If you disable the automatic generation of sequence numbers, you must specify the sequence number for each entry. You can specify sequence values in any increment. If you specify increments of one, you cannot insert additional entries into the list; if you choose very large increments, you might run out of values.

You do not need to specify a sequence number when removing a configuration entry. **Show** commands include the sequence numbers in their output.

Before using a prefix list in a command, you must set up the prefix list. Beginning in privileged EXEC mode, follow these steps to create a prefix list or to add an entry to a prefix list:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip prefix-list** *list-name* [**seq** *seq-value*] **deny** \| **permit** *network*/*len* [**ge** *ge-value*] [**le** *le-value*]	Create a prefix list with an optional sequence number to **deny** or **permit** access for matching conditions. You must enter at least one **permit** or **deny** clause. • *network*/*len* is the network number and length (in bits) of the network mask. • (Optional) **ge** and **le** values specify the range of the prefix length to be matched.The specified *ge-value* and *le-value* must satisfy this condition: *len* < *ge-value* < *le-value* < 32
Step 3	**ip prefix-list** *list-name* **seq** *seq-value* **deny** \| **permit** *network*/*len* [**ge** *ge-value*] [**le** *le-value*]	(Optional) Add an entry to a prefix list, and assign a sequence number to the entry.

	Command	Purpose
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show ip prefix list** [**detail** \| **summary**] *name* [*network*/*len*] [**seq** *seq-num*] [**longer**] [**first-match**]	Verify the configuration by displaying information about a prefix list or prefix list entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To delete a prefix list and all of its entries, use the **no ip prefix-list** *list-name* global configuration command. To delete an entry from a prefix list, use the **no ip prefix-list seq** *seq-value* global configuration command. To disable automatic generation of sequence numbers, use the **no ip prefix-list sequence number** command; to reenable automatic generation, use the **ip prefix-list sequence number** command. To clear the hit-count table of prefix list entries, use the **clear ip prefix-list** privileged EXEC command.

Configuring BGP Community Filtering

One way that BGP controls the distribution of routing information based on the value of the COMMUNITIES attribute. The attribute is a way to groups destinations into communities and to apply routing decisions based on the communities. This method simplifies configuration of a BGP speaker to control distribution of routing information.

A *community* is a group of destinations that share some common attribute. Each destination can belong to multiple communities. AS administrators can define to which communities a destination belongs. By default, all destinations belong to the general Internet community. The community is identified by the COMMUNITIES attribute, an optional, transitive, global attribute in the numerical range from 1 to 4294967200. These are some predefined, well-known communities:

- **internet**—Advertise this route to the Internet community. All routers belong to it.
- **no-export**—Do not advertise this route to EBGP peers.
- **no-advertise**—Do not advertise this route to any peer (internal or external).
- **local-as**—Do not advertise this route to peers outside the local autonomous system.

Based on the community, you can control which routing information to accept, prefer, or distribute to other neighbors. A BGP speaker can set, append, or modify the community of a route when learning, advertising, or redistributing routes. When routes are aggregated, the resulting aggregate has a COMMUNITIES attribute that contains all communities from all the initial routes.

You can use community lists to create groups of communities to use in a match clause of a route map. As with an access list, a series of community lists can be created. Statements are checked until a match is found. As soon as one statement is satisfied, the test is concluded.

To set the COMMUNITIES attribute and match clauses based on communities, see the **match community-list** and **set community** route-map configuration commands in the "Using Route Maps to Redistribute Routing Information" section on page 32-73.

By default, no COMMUNITIES attribute is sent to a neighbor. You can specify that the COMMUNITIES attribute be sent to the neighbor at an IP address by using the **neighbor send-community** router configuration command.

Beginning in privileged EXEC mode, follow these steps to create and to apply a community list:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip community-list** *community-list-number* {**permit** \| **deny**} *community-number*	Create a community list, and assign it a number. • The *community-list-number* is an integer from 1 to 99 that identifies one or more permit or deny groups of communities. • The *community-number* is the number configured by a **set community** route-map configuration command.
Step 3	**router bgp** *autonomous-system*	Enter BGP router configuration mode.
Step 4	**neighbor** {*ip-address* \| *peer-group name*} **send-community**	Specify that the COMMUNITIES attribute be sent to the neighbor at this IP address.
Step 5	**set comm-list** *list-num* **delete**	(Optional) Remove communities from the community attribute of an inbound or outbound update that match a standard or extended community list specified by a route map.
Step 6	**exit**	Return to global configuration mode.
Step 7	**ip bgp-community new-format**	(Optional) Display and parse BGP communities in the format AA:NN. A BGP community is displayed in a two-part format 2 bytes long. The Cisco default community format is in the format NNAA. In the most recent RFC for BGP, a community takes the form AA:NN, where the first part is the AS number and the second part is a 2-byte number.
Step 8	**end**	Return to privileged EXEC mode.
Step 9	**show ip bgp community**	Verify the configuration.
Step 10	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Configuring BGP Neighbors and Peer Groups

Often many BGP neighbors are configured with the same update policies (that is, the same outbound route maps, distribute lists, filter lists, update source, and so on). Neighbors with the same update policies can be grouped into peer groups to simplify configuration and to make updating more efficient. When you have configured many peers, we recommend this approach.

To configure a BGP peer group, you create the peer group, assign options to the peer group, and add neighbors as peer group members. You configure the peer group by using the **neighbor** router configuration commands. By default, peer group members inherit all the configuration options of the peer group, including the remote-as (if configured), version, update-source, out-route-map, out-filter-list, out-dist-list, minimum-advertisement-interval, and next-hop-self. All peer group members also inherit changes made to the peer group. Members can also be configured to override the options that do not affect outbound updates.

To assign configuration options to an individual neighbor, specify any of these router configuration commands by using the neighbor IP address. To assign the options to a peer group, specify any of the commands by using the peer group name. You can disable a BGP peer or peer group without removing all the configuration information by using the **neighbor shutdown** router configuration command.

Beginning in privileged EXEC mode, use these commands to configure BGP peers:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**router bgp** *autonomous-system*	Enter BGP router configuration mode.
Step 3	**neighbor** *peer-group-name* **peer-group**	Create a BGP peer group.
Step 4	**neighbor** *ip-address* **peer-group** *peer-group-name*	Make a BGP neighbor a member of the peer group.
Step 5	**neighbor** {*ip-address* \| *peer-group-name*} **remote-as** *number*	Specify a BGP neighbor. If a peer group is not configured with a **remote-as** *number*, use this command to create peer groups containing EBGP neighbors. The range is 1 to 65535.
Step 6	**neighbor** {*ip-address* \| *peer-group-name*} **description** *text*	(Optional) Associate a description with a neighbor.
Step 7	**neighbor** {*ip-address* \| *peer-group-name*} **default-originate** [**route-map** *map-name*]	(Optional) Allow a BGP speaker (the local router) to send the default route 0.0.0.0 to a neighbor for use as a default route.
Step 8	**neighbor** {*ip-address* \| *peer-group-name*} **send-community**	(Optional) Specify that the COMMUNITIES attribute be sent to the neighbor at this IP address.
Step 9	**neighbor** {*ip-address* \| *peer-group-name*} **update-source** *interface*	(Optional) Allow internal BGP sessions to use any operational interface for TCP connections.
Step 10	**neighbor** {*ip-address* \| *peer-group-name*} **ebgp-multihop**	(Optional) Allow BGP sessions, even when the neighbor is not on a directly connected segment. The multihop session is not established if the only route to the multihop peer's address is the default route (0.0.0.0).
Step 11	**neighbor** {*ip-address* \| *peer-group-name*} **local-as** *number*	(Optional) Specify an AS number to use as the local AS. The range is 1 to 65535.
Step 12	**neighbor** {*ip-address* \| *peer-group-name*} **advertisement-interval** *seconds*	(Optional) Set the minimum interval between sending BGP routing updates.
Step 13	**neighbor** {*ip-address* \| *peer-group-name*} **maximum-prefix** *maximum* [*threshold*]	(Optional) Control how many prefixes can be received from a neighbor. The range is 1 to 4294967295. The *threshold* (optional) is the percentage of maximum at which a warning message is generated. The default is 75 percent.
Step 14	**neighbor** {*ip-address* \| *peer-group-name*} **next-hop-self**	(Optional) Disable next-hop processing on the BGP updates to a neighbor.
Step 15	**neighbor** {*ip-address* \| *peer-group-name*} **password** *string*	(Optional) Set MD5 authentication on a TCP connection to a BGP peer. The same password must be configured on both BGP peers, or the connection between them is not made.
Step 16	**neighbor** {*ip-address* \| *peer-group-name*} **route-map** *map-name* {**in** \| **out**}	(Optional) Apply a route map to incoming or outgoing routes.
Step 17	**neighbor** {*ip-address* \| *peer-group-name*} **send-community**	(Optional) Specify that the COMMUNITIES attribute be sent to the neighbor at this IP address.

	Command	Purpose
Step 18	**neighbor** {*ip-address* \| *peer-group-name*} **timers** *keepalive holdtime*	(Optional) Set timers for the neighbor or peer group. • The *keepalive* interval is the time within which keepalive messages are sent to peers. The range is 1 to 4294967295 seconds; the default is 60. • The *holdtime* is the interval after which a peer is declared inactive after not receiving a keepalive message from it. The range is 1 to 4294967295 seconds; the default is 180.
Step 19	**neighbor** {*ip-address* \| *peer-group-name*} **weight** *weight*	(Optional) Specify a weight for all routes from a neighbor.
Step 20	**neighbor** {*ip-address* \| *peer-group-name*} **distribute-list** {*access-list-number* \| *name*} {**in** \| **out**}	(Optional) Filter BGP routing updates to or from neighbors, as specified in an access list.
Step 21	**neighbor** {*ip-address* \| *peer-group-name*} **filter-list** *access-list-number* {**in** \| **out** \| **weight** *weight*}	(Optional) Establish a BGP filter.
Step 22	**neighbor** {*ip-address* \| *peer-group-name*} **version** *value*	(Optional) Specify the BGP version to use when communicating with a neighbor.
Step 23	**neighbor** {*ip-address* \| *peer-group-name*} **soft-reconfiguration inbound**	(Optional) Configure the software to start storing received updates.
Step 24	**end**	Return to privileged EXEC mode.
Step 25	**show ip bgp neighbors**	Verify the configuration.
Step 26	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable an existing BGP neighbor or neighbor peer group, use the **neighbor shutdown** router configuration command. To enable a previously existing neighbor or neighbor peer group that had been disabled, use the **no neighbor shutdown** router configuration command.

Configuring Aggregate Addresses

Classless interdomain routing (CIDR) enables you to create aggregate routes (or *supernets*) to minimize the size of routing tables. You can configure aggregate routes in BGP either by redistributing an aggregate route into BGP or by creating an aggregate entry in the BGP routing table. An aggregate address is added to the BGP table when there is at least one more specific entry in the BGP table.

Beginning in privileged EXEC mode, use these commands to create an aggregate address in the routing table:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**router bgp** *autonomous-system*	Enter BGP router configuration mode.
Step 3	**aggregate-address** *address mask*	Create an aggregate entry in the BGP routing table. The aggregate route is advertised as coming from the AS, and the atomic aggregate attribute is set to indicate that information might be missing.

	Command	Purpose
Step 4	**aggregate-address** *address mask* **as-set**	(Optional) Generate AS set path information. This command creates an aggregate entry following the same rules as the previous command, but the advertised path will be an AS_SET consisting of all elements contained in all paths. Do not use this keyword when aggregating many paths because this route must be continually withdrawn and updated.
Step 5	**aggregate-address** *address-mask* **summary-only**	(Optional) Advertise summary addresses only.
Step 6	**aggregate-address** *address mask* **suppress-map** *map-name*	(Optional) Suppress selected, more specific routes.
Step 7	**aggregate-address** *address mask* **advertise-map** *map-name*	(Optional) Generate an aggregate based on conditions specified by the route map.
Step 8	**aggregate-address** *address mask* **attribute-map** *map-name*	(Optional) Generate an aggregate with attributes specified in the route map.
Step 9	**end**	Return to privileged EXEC mode.
Step 10	**show ip bgp neighbors** [**advertised-routes**]	Verify the configuration.
Step 11	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To delete an aggregate entry, use the **no aggregate-address** *address mask* router configuration command. To return options to the default values, use the command with keywords.

Configuring Routing Domain Confederations

One way to reduce the IBGP mesh is to divide an autonomous system into multiple subautonomous systems and to group them into a single confederation that appears as a single autonomous system. Each autonomous system is fully meshed within itself and has a few connections to other autonomous systems in the same confederation. Even though the peers in different autonomous systems have EBGP sessions, they exchange routing information as if they were IBGP peers. Specifically, the next hop, MED, and local preference information is preserved. You can then use a single IGP for all of the autonomous systems.

To configure a BGP confederation, you must specify a confederation identifier that acts as the autonomous system number for the group of autonomous systems.

Beginning in privileged EXEC mode, use these commands to configure a BGP confederation:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**router bgp** *autonomous-system*	Enter BGP router configuration mode.
Step 3	**bgp confederation identifier** *autonomous-system*	Configure a BGP confederation identifier.
Step 4	**bgp confederation peers** *autonomous-system* [*autonomous-system ...*]	Specify the autonomous systems that belong to the confederation and that will be treated as special EBGP peers.
Step 5	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 6	**show ip bgp neighbor** **show ip bgp network**	Verify the configuration.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Configuring BGP Route Reflectors

BGP requires that all of the IBGP speakers be fully meshed. When a router receives a route from an external neighbor, it must advertise it to all internal neighbors. To prevent a routing information loop, all IBPG speakers must be connected. The internal neighbors do not send routes learned from internal neighbors to other internal neighbors.

With route reflectors, all IBGP speakers need not be fully meshed because another method is used to pass learned routes to neighbors. When you configure an internal BGP peer to be a *route reflector*, it is responsible for passing IBGP learned routes to a set of IBGP neighbors. The internal peers of the route reflector are divided into two groups: *client peers* and *nonclient peers (*all the other routers in the autonomous system). A route reflector reflects routes between these two groups. The route reflector and its client peers form a *cluster.* The nonclient peers must be fully meshed with each other, but the client peers need not be fully meshed. The clients in the cluster do not communicate with IBGP speakers outside their cluster.

When the route reflector receives an advertised route, it takes one of these actions, depending on the neighbor:

- A route from an external BGP speaker is advertised to all clients and nonclient peers.
- A route from a nonclient peer is advertised to all clients.
- A route from a client is advertised to all clients and nonclient peers. Hence, the clients need not be fully meshed.

Usually a cluster of clients have a single route reflector, and the cluster is identified by the route reflector router ID. To increase redundancy and to avoid a single point of failure, a cluster might have more than one route reflector. In this case, all route reflectors in the cluster must be configured with the same 4-byte cluster ID so that a route reflector can recognize updates from route reflectors in the same cluster. All the route reflectors serving a cluster should be fully meshed and should have identical sets of client and nonclient peers.

Beginning in privileged EXEC mode, use these commands to configure a route reflector and clients:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**router bgp** *autonomous-system*	Enter BGP router configuration mode.
Step 3	**neighbor** *ip-address* \| *peer-group-name* **route-reflector-client**	Configure the local router as a BGP route reflector and the specified neighbor as a client.
Step 4	**bgp cluster-id** *cluster-id*	(Optional) Configure the cluster ID if the cluster has more than one route reflector.
Step 5	**no bgp client-to-client reflection**	(Optional) Disable client-to-client route reflection. By default, the routes from a route reflector client are reflected to other clients. However, if the clients are fully meshed, the route reflector does not need to reflect routes to clients.

	Command	Purpose
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show ip bgp**	Verify the configuration. Display the originator ID and the cluster-list attributes.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Configuring Route Dampening

Route flap dampening is a BGP feature designed to minimize the propagation of flapping routes across an internetwork. A route is considered to be flapping when it is repeatedly available, then unavailable, then available, then unavailable, and so on. When route dampening is enabled, a numeric *penalty* value is assigned to a route when it flaps. When a route's accumulated penalties reach a configurable limit, BGP suppresses advertisements of the route, even if the route is running. The *reuse limit* is a configurable value that is compared with the penalty. If the penalty is less than the reuse limit, a suppressed route that is up is advertised again.

Dampening is not applied to routes that are learned by IBGP. This policy prevents the IBGP peers from having a higher penalty for routes external to the AS.

Beginning in privileged EXEC mode, use these commands to configure BGP route dampening:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**router bgp** *autonomous-system*	Enter BGP router configuration mode.
Step 3	**bgp dampening**	Enable BGP route dampening.
Step 4	**bgp dampening** *half-life reuse suppress max-suppress* [**route-map** *map*]	(Optional) Change the default values of route dampening factors.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show ip bgp flap-statistics** [{**regexp** *regexp*} \| {**filter-list** *list*} \| {*address mask* [**longer-prefix**]}]	(Optional) Monitor the flaps of all paths that are flapping. The statistics are deleted when the route is not suppressed and is stable.
Step 7	**show ip bgp dampened-paths**	(Optional) Display the dampened routes, including the time remaining before they are suppressed.
Step 8	**clear ip bgp flap-statistics** [{**regexp** *regexp*} \| {**filter-list** *list*} \| {*address mask* [**longer-prefix**]}	(Optional) Clear BGP flap statistics to make it less likely that a route will be dampened.
Step 9	**clear ip bgp dampening**	(Optional) Clear route dampening information, and unsuppress the suppressed routes.
Step 10	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable flap dampening, use the **no bgp dampening** router configuration command without keywords. To set dampening factors back to the default values, use the **no bgp dampening** router configuration command with values.

Monitoring and Maintaining BGP

You can remove all contents of a particular cache, table, or database. This might be necessary when the contents of the particular structure have become or are suspected to be invalid.

You can display specific statistics, such as the contents of BGP routing tables, caches, and databases. You can use the information to get resource utilization and solve network problems. You can also display information about node reachability and discover the routing path your device's packets are taking through the network.

Table 32-8 lists the privileged EXEC commands for clearing and displaying BGP. For explanations of the display fields, see the *Cisco IOS IP Command Reference, Volume 2 of 3: Routing Protocols, Release 12.2*.

Table 32-11 IP BGP Clear and Show Commands

Command	Purpose
clear ip bgp *address*	Reset a particular BGP connection.
clear ip bgp *	Reset all BGP connections.
clear ip bgp peer-group *tag*	Remove all members of a BGP peer group.
show ip bgp *prefix*	Display peer groups and peers not in peer groups to which the prefix has been advertised. Also display prefix attributes such as the next hop and the local prefix.
show ip bgp cidr-only	Display all BGP routes that contain subnet and supernet network masks.
show ip bgp community [*community-number*] [**exact**]	Display routes that belong to the specified communities.
show ip bgp community-list *community-list-number* [**exact-match**]	Display routes that are permitted by the community list.
show ip bgp filter-list *access-list-number*	Display routes that are matched by the specified AS path access list.
show ip bgp inconsistent-as	Display the routes with inconsistent originating autonomous systems.
show ip bgp regexp *regular-expression*	Display the routes that have an AS path that matches the specified regular expression entered on the command line.
show ip bgp	Display the contents of the BGP routing table.
show ip bgp neighbors [*address*]	Display detailed information on the BGP and TCP connections to individual neighbors.
show ip bgp neighbors [*address*] [**advertised-routes** \| **dampened-routes** \| **flap-statistics** \| **paths** *regular-expression* \| **received-routes** \| **routes**]	Display routes learned from a particular BGP neighbor.
show ip bgp paths	Display all BGP paths in the database.
show ip bgp peer-group [*tag*] [**summary**]	Display information about BGP peer groups.
show ip bgp summary	Display the status of all BGP connections.

You can also enable the logging of messages generated when a BGP neighbor resets, comes up, or goes down by using the **bgp log-neighbor changes** router configuration command.

Configuring Multi-VRF CE

Virtual Private Networks (VPNs) provide a secure way for customers to share bandwidth over an ISP backbone network. A VPN is a collection of sites sharing a common routing table. A customer site is connected to the service-provider network by one or more interfaces, and the service provider associates each interface with a VPN routing table, called a VPN routing/forwarding (VRF) table.

The switch supports multiple VPN routing/forwarding (multi-VRF) instances in customer edge (CE) devices (multi-VRF CE). Multi-VRF CE allows a service provider to support two or more VPNs with overlapping IP addresses.

Note The switch does not use Multiprotocol Label Switching (MPLS) to support VPNs. For information about MPLS VRF, refer to the *Cisco IOS Switching Services Configuration Guide, Release 12.2*.

These sections contain this information:

- Understanding Multi-VRF CE, page 32-59
- Default Multi-VRF CE Configuration, page 32-61
- Multi-VRF CE Configuration Guidelines, page 32-61
- Configuring VRFs, page 32-62
- Configuring a VPN Routing Session, page 32-63
- Configuring BGP PE to CE Routing Sessions, page 32-64
- Multi-VRF CE Configuration Example, page 32-64
- Displaying Multi-VRF CE Status, page 32-68

Understanding Multi-VRF CE

Multi-VRF CE is a feature that allows a service provider to support two or more VPNs, where IP addresses can be overlapped among the VPNs. Multi-VRF CE uses input interfaces to distinguish routes for different VPNs and forms virtual packet-forwarding tables by associating one or more Layer 3 interfaces with each VRF. Interfaces in a VRF can be either physical, such as Ethernet ports, or logical, such as VLAN SVIs, but an interface cannot belong to more than one VRF at any time.

Note Multi-VRF CE interfaces must be Layer 3 interfaces.

Multi-VRF CE includes these devices:

- Customer edge (CE) devices provide customers access to the service-provider network over a data link to one or more provider edge routers. The CE device advertises the site's local routes to the router and learns the remote VPN routes from it. The Cisco ME 3400 switch can be a CE.
- Provider edge (PE) routers exchange routing information with CE devices by using static routing or a routing protocol such as BGP, RIPv2, OSPF, or EIGRP. The PE is only required to maintain VPN routes for those VPNs to which it is directly attached, eliminating the need for the PE to maintain all of the service-provider VPN routes. Each PE router maintains a VRF for each of its directly connected sites. Multiple interfaces on a PE router can be associated with a single VRF if all of these

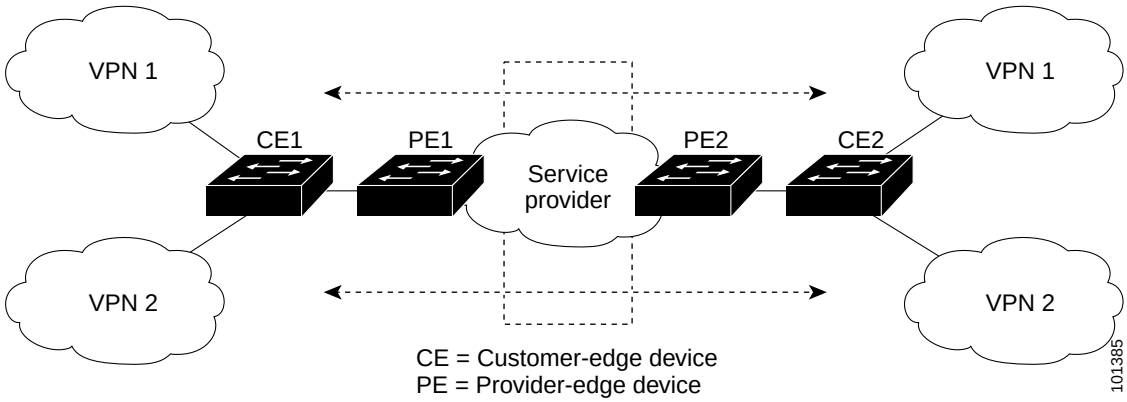
sites participate in the same VPN. Each VPN is mapped to a specified VRF. After learning local VPN routes from CEs, a PE router exchanges VPN routing information with other PE routers by using internal BGP (IBPG).

- Provider routers or core routers are any routers in the service provider network that do not attach to CE devices.

With multi-VRF CE, multiple customers can share one CE, and only one physical link is used between the CE and the PE. The shared CE maintains separate VRF tables for each customer and switches or routes packets for each customer based on its own routing table. Multi-VRF CE extends limited PE functionality to a CE device, giving it the ability to maintain separate VRF tables to extend the privacy and security of a VPN to the branch office.

Figure 32-5 shows a configuration using Cisco ME 3400 switches as multiple virtual CEs. This scenario is suited for customers who have low bandwidth requirements for their VPN service, for example, small companies. In this case, multi-VRF CE support is required in the Cisco ME switches. Because multi-VRF CE is a Layer 3 feature, each interface in a VRF must be a Layer 3 interface.

Figure 32-5 Switches Acting as Multiple Virtual CEs



When the CE switch receives a command to add a Layer 3 interface to a VRF, it sets up the appropriate mapping between the VLAN ID and the policy label (PL) in multi-VRF-CE-related data structures and adds the VLAN ID and PL to the VLAN database.

When multi-VRF CE is configured, the Layer 3 forwarding table is conceptually partitioned into two sections:

- The multi-VRF CE routing section contains the routes from different VPNs.
- The global routing section contains routes to non-VPN networks, such as the Internet.

VLAN IDs from different VRFs are mapped into different policy labels, which are used to distinguish the VRFs during processing. If no route is found in the multi-VRF CE section of the Layer 3 forwarding table, the global routing section is used to determine the forwarding path. For each new VPN route learned, the Layer 3 setup function retrieves the policy label by using the VLAN ID of the ingress port and inserts the policy label and new route to the multi-VRF CE routing section. If the packet is received from a routed port, the port internal VLAN ID number is used; if the packet is received from an SVI, the VLAN number is used.

This is the packet-forwarding process in a multi-VRF-CE-enabled network:

- When the switch receives a packet from a VPN, the switch looks up the routing table based on the input policy label number. When a route is found, the switch forwards the packet to the PE.
- When the ingress PE receives a packet from the CE, it performs a VRF lookup. When a route is found, the router adds a corresponding MPLS label to the packet and sends it to the MPLS network.
- When an egress PE receives a packet from the network, it strips the label and uses the label to identify the correct VPN routing table. Then it performs the normal route lookup. When a route is found, it forwards the packet to the correct adjacency.
- When a CE receives a packet from an egress PE, it uses the input policy label to look up the correct VPN routing table. If a route is found, it forwards the packet within the VPN.

To configure VRF, you create a VRF table and specify the Layer 3 interface associated with the VRF. Then configure the routing protocols in the VPN and between the CE and the PE. BGP is the preferred routing protocol used to distribute VPN routing information across the provider's backbone. The multi-VRF CE network has three major components:

- VPN route target communities—lists of all other members of a VPN community. You need to configure VPN route targets for each VPN community member.
- Multiprotocol BGP peering of VPN community PE routers—propagates VRF reachability information to all members of a VPN community. You need to configure BGP peering in all PE routers within a VPN community.
- VPN forwarding—transports all traffic between all VPN community members across a VPN service-provider network.

Default Multi-VRF CE Configuration

Table 32-12 shows the default VRF configuration.

Table 32-12 Default VRF Configuration

Feature	Default Setting
VRF	Disabled. No VRFs are defined.
Maps	No import maps, export maps, or route maps are defined.
VRF maximum routes	Fast Ethernet switches: 8000 Gigabit Ethernet switches: 12000.
Forwarding table	The default for an interface is the global routing table.

Multi-VRF CE Configuration Guidelines

These are considerations when configuring VRF in your network:

- A switch with multi-VRF CE is shared by multiple customers, and each customer has its own routing table.
- Because customers use different VRF tables, the same IP addresses can be reused. Overlapped IP addresses are allowed in different VPNs.

- Multi-VRF CE lets multiple customers share the same physical link between the PE and the CE. Trunk ports with multiple VLANs separate packets among customers. Each customer has its own VLAN.
- Multi-VRF CE does not support all MPLS-VRF functionality. It does not support label exchange, LDP adjacency, or labeled packets.
- For the PE router, there is no difference between using multi-VRF CE or using multiple CEs. In Figure 32-5, multiple virtual Layer 3 interfaces are connected to the multi-VRF CE device.
- The switch supports configuring VRF by using physical ports, VLAN SVIs, or a combination of both. The SVIs can be connected through an access port or a trunk port.
- A customer can use multiple VLANs as long as they do not overlap with those of other customers. A customer's VLANs are mapped to a specific routing table ID that is used to identify the appropriate routing tables stored on the switch.
- The switch supports one global network and up to 26 VRFs.
- Most routing protocols (BGP, OSPF, RIP, EIGRP, and static routing) can be used between the CE and the PE. However, we recommend using external BGP (EBGP) for these reasons:
 - BGP does not require multiple algorithms to communicate with multiple CEs.
 - BGP is designed for passing routing information between systems run by different administrations.
 - BGP makes it easy to pass attributes of the routes to the CE.
- Multi-VRF CE does not affect the packet switching rate.
- VPN multicast is not supported.
- If no VRFs are configured then 104 policies can be configured.
- If even one VRF is configured than 41 policies can be configured.
- If more than 41 policies are configured then VRF cannot be configured.
- VRF and private VLANs are mutually-exclusive. You cannot enable VRF on a private VLAN. Similarly, you cannot enable private VLAN on a VLAN with VRF configured on the VLAN interface.
- VRF and policy-based routing (PBR) are mutually-exclusive on a switch interface. You cannot enable VRF when PBR is enabled on an interface. In contrast, you cannot enable PBR when VRF is enabled on an interface.

Configuring VRFs

Beginning in privileged EXEC mode, follow these steps to configure one or more VRFs. For complete syntax and usage information for the commands, refer to the switch command reference for this release and the *Cisco IOS Switching Services Command Reference, Release 12.2*.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip routing**	Enable IP routing.
Step 3	**ip vrf** *vrf-name*	Name the VRF, and enter VRF configuration mode.

	Command	Purpose
Step 4	**rd** *route-distinguisher*	Create a VRF table by specifying a route distinguisher. Enter either an AS number and an arbitrary number (xxx:y) or an IP address and arbitrary number (A.B.C.D:y)
Step 5	**route-target** {**export** \| **import** \| **both**} *route-target-ext-community*	Create a list of import, export, or import and export route target communities for the specified VRF. Enter either an AS system number and an arbitrary number (xxx:y) or an IP address and an arbitrary number (A.B.C.D:y). The *route-target-ext-community* should be the same as the *route-distinguisher* entered in Step 4.
Step 6	**import map** *route-map*	(Optional) Associate a route map with the VRF.
Step 7	**interface** *interface-id*	Specify the Layer 3 interface to be associated with the VRF, and enter interface configuration mode. The interface can be a routed port or SVI.
Step 8	**no shutdown**	Enable the interface if necessary. By default, UNIs are disabled and NNIs are enabled.
Step 9	**ip vrf forwarding** *vrf-name*	Associate the VRF with the Layer 3 interface.
Step 10	**end**	Return to privileged EXEC mode.
Step 11	**show ip vrf** [**brief** \| **detail** \| **interfaces**] [*vrf-name*]	Verify the configuration. Display information about the configured VRFs.
Step 12	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no ip vrf** *vrf-name* global configuration command to delete a VRF and to remove all interfaces from it. Use the **no ip vrf forwarding** interface configuration command to remove an interface from the VRF.

Configuring a VPN Routing Session

Routing within the VPN can be configured with any supported routing protocol (RIP, OSPF, EIGRP, or BGP) or with static routing. The configuration shown here is for OSPF, but the process is the same for other protocols.

Beginning in privileged EXEC mode, follow these steps to configure OSPF in the VPN:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**router ospf** *process-id* **vrf** *vrf-name*	Enable OSPF routing, specify a VPN forwarding table, and enter router configuration mode.
Step 3	**log-adjacency-changes**	(Optional) Log changes in the adjacency state. This is the default state.
Step 4	**redistribute bgp** *autonomous-system-number* **subnets**	Set the switch to redistribute information from the BGP network to the OSPF network.
Step 5	**network** *network-number* **area** *area-id*	Define a network address and mask on which OSPF runs and the area ID for that network address.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show ip ospf** *process-id*	Verify the configuration of the OSPF network.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no router ospf** *process-id* **vrf** *vrf-name* global configuration command to disassociate the VPN forwarding table from the OSPF routing process.

Configuring BGP PE to CE Routing Sessions

Beginning in privileged EXEC mode, follow these steps to configure a BGP PE to CE routing session:

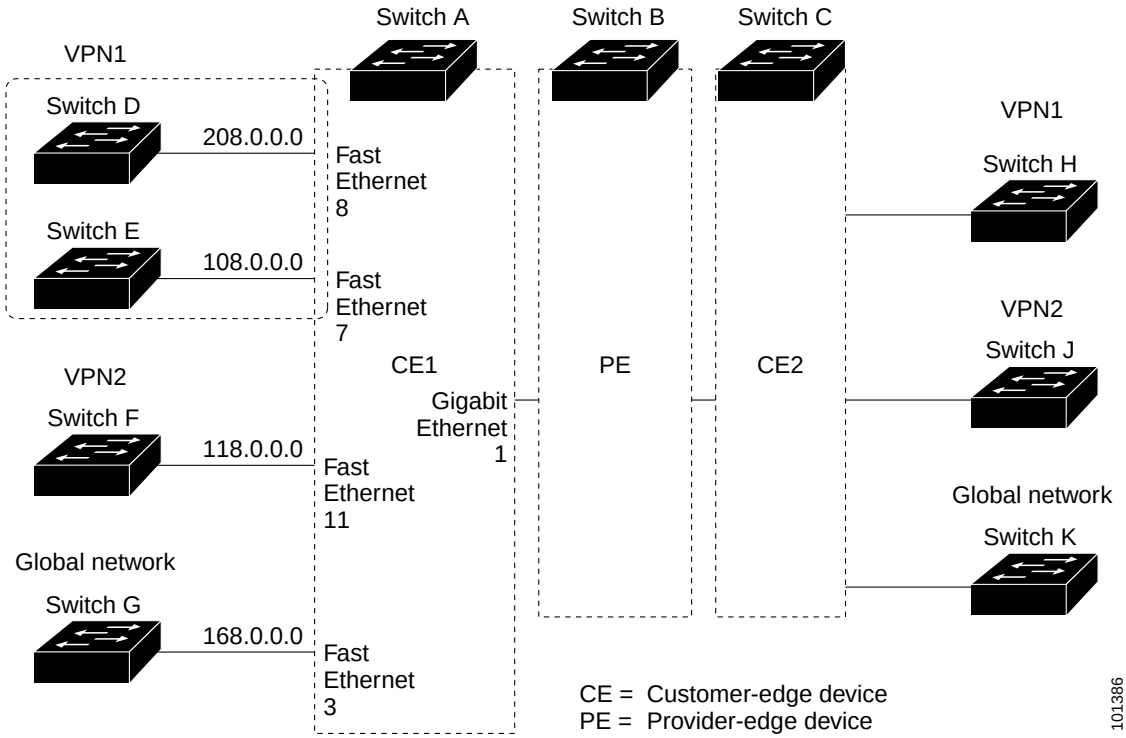
	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**router bgp** *autonomous-system-number*	Configure the BGP routing process with the AS number passed to other BGP routers, and enter router configuration mode.
Step 3	**network** *network-number* **mask** *network-mask*	Specify a network and mask to announce using BGP.
Step 4	**redistribute ospf** *process-id* **match internal**	Set the switch to redistribute OSPF internal routes.
Step 5	**network** *network-number* **area** *area-id*	Define a network address and mask on which OSPF runs and the area ID for that network address.
Step 6	**address-family ipv4 vrf** *vrf-name*	Define BGP parameters for PE to CE routing sessions, and enter VRF address-family mode.
Step 7	**neighbor** *address* **remote-as** *as-number*	Define a BGP session between PE and CE routers.
Step 8	**neighbor** *address* **activate**	Activate the advertisement of the IPv4 address family.
Step 9	**end**	Return to privileged EXEC mode.
Step 10	**show ip bgp** [**ipv4**] [**neighbors**]	Verify BGP configuration.
Step 11	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no router bgp** *autonomous-system-number* global configuration command to delete the BGP routing process. Use the command with keywords to delete routing characteristics.

Multi-VRF CE Configuration Example

Figure 32-6 is a simplified example of the physical connections in a network similar to that in Figure 32-5. OSPF is the protocol used in VPN1, VPN2, and the global network. BGP is used in the CE to PE connections. The examples following the illustration show how to configure a Cisco ME 3400 switch as CE Switch A, and the VRF configuration for customer switches D and F. Commands for configuring CE Switch C and the other customer switches are not included but would be similar. The example also includes commands for configuring traffic to Switch A for a Catalyst 6000 or Catalyst 6500 switch acting as a PE router.

Figure 32-6 Multi-VRF CE Configuration Example



Configuring Switch A

On Switch A, enable routing and configure VRF.

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# ip routing
Switch(config)# ip vrf v11
Switch(config-vrf)# rd 800:1
Switch(config-vrf)# route-target export 800:1
Switch(config-vrf)# route-target import 800:1
Switch(config-vrf)# exit
Switch(config)# ip vrf v12
Switch(config-vrf)# rd 800:2
Switch(config-vrf)# route-target export 800:2
Switch(config-vrf)# route-target import 800:2
Switch(config-vrf)# exit
```

Configure the loopback and physical interfaces on Switch A. Gigabit Ethernet port 1 is a trunk connection to the PE. Fast Ethernet ports 8 and 11 connect to VPNs:

```
Switch(config)# interface loopback1
Switch(config-if)# ip vrf forwarding v11
Switch(config-if)# ip address 8.8.1.8 255.255.255.0
Switch(config-if)# exit

Switch(config)# interface loopback2
Switch(config-if)# ip vrf forwarding v12
Switch(config-if)# ip address 8.8.2.8 255.255.255.0
Switch(config-if)# exit
```

```
Switch(config)# interface gigabitethernet1/0/5
Switch(config-if)# switchport trunk encapsulation dot1q
Switch(config-if)# switchport mode trunk
Switch(config-if)# no ip address
Switch(config-if)# exit

Switch(config)# interface fastethernet1/0/8
Switch(config-if)# no shutdown
Switch(config-if)# switchport access vlan 208
Switch(config-if)# no ip address
Switch(config-if)# exit

Switch(config)# interface fastethernet1/0/11
Switch(config-if)# no shutdown
Switch(config-if)# switchport trunk encapsulation dot1q
Switch(config-if)# switchport mode trunk
Switch(config-if)# no ip address
Switch(config-if)# exit
```

Configure the VLANs used on Switch A. VLAN 10 is used by VRF 11 between the CE and the PE. VLAN 20 is used by VRF 12 between the CE and the PE. VLANs 118 and 208 are used for the VPNs that include Switch F and Switch D, respectively:

```
Switch(config)# interface vlan10
Switch(config-if)# ip vrf forwarding v11
Switch(config-if)# ip address 38.0.0.8 255.255.255.0
Switch(config-if)# exit

Switch(config)# interface vlan20
Switch(config-if)# ip vrf forwarding v12
Switch(config-if)# ip address 83.0.0.8 255.255.255.0
Switch(config-if)# exit

Switch(config)# interface vlan118
Switch(config-if)# ip vrf forwarding v12
Switch(config-if)# ip address 118.0.0.8 255.255.255.0
Switch(config-if)# exit

Switch(config)# interface vlan208
Switch(config-if)# ip vrf forwarding v11
Switch(config-if)# ip address 208.0.0.8 255.255.255.0
Switch(config-if)# exit
```

Configure OSPF routing in VPN1 and VPN2.

```
Switch(config)# router ospf 1 vrf vl1
Switch(config-router)# redistribute bgp 800 subnets
Switch(config-router)# network 208.0.0.0 0.0.0.255 area 0
Switch(config-router)# exit
Switch(config)# router ospf 2 vrf vl2
Switch(config-router)# redistribute bgp 800 subnets
Switch(config-router)# network 118.0.0.0 0.0.0.255 area 0
Switch(config-router)# exit
```

Configure BGP for CE to PE routing.

```
Switch(config)# router bgp 800
Switch(config-router)# address-family ipv4 vrf vl2
Switch(config-router-af)# redistribute ospf 2 match internal
Switch(config-router-af)# neighbor 83.0.0.3 remote-as 100
Switch(config-router-af)# neighbor 83.0.0.3 activate
Switch(config-router-af)# network 8.8.2.0 mask 255.255.255.0
Switch(config-router-af)# exit
```

```
Switch(config-router)# address-family ipv4 vrf vl1
Switch(config-router-af)# redistribute ospf 1 match internal
Switch(config-router-af)# neighbor 38.0.0.3 remote-as 100
Switch(config-router-af)# neighbor 38.0.0.3 activate
Switch(config-router-af)# network 8.8.1.0 mask 255.255.255.0
Switch(config-router-af)# end
```

Configuring Switch D

Switch D belongs to VPN 1. Configure the connection to Switch A by using these commands.

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# ip routing
Switch(config)# interface fastethernet1/0/2
Switch(config-if)# no shutdown
Switch(config-if)# no switchport
Switch(config-if)# ip address 208.0.0.20 255.255.255.0
Switch(config-if)# exit

Switch(config)# router ospf 101
Switch(config-router)# network 208.0.0.0 0.0.0.255 area 0
Switch(config-router)# end
```

Configuring Switch F

Switch F belongs to VPN 2. Configure the connection to Switch A by using these commands.

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# ip routing
Switch(config)# interface fastethernet1/0/1
Switch(config-if)# no shutdown
Switch(config-if)# switchport trunk encapsulation dot1q
Switch(config-if)# switchport mode trunk
Switch(config-if)# no ip address
Switch(config-if)# exit

Switch(config)# interface vlan118
Switch(config-if)# ip address 118.0.0.11 255.255.255.0
Switch(config-if)# exit

Switch(config)# router ospf 101
Switch(config-router)# network 118.0.0.0 0.0.0.255 area 0
Switch(config-router)# end
```

Configuring the PE Switch B

On Switch B (the PE router), these commands configure only the connections to the CE device, Switch A.

```
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# ip vrf v1
Router(config-vrf)# rd 100:1
Router(config-vrf)# route-target export 100:1
Router(config-vrf)# route-target import 100:1
Router(config-vrf)# exit

Router(config)# ip vrf v2
Router(config-vrf)# rd 100:2
Router(config-vrf)# route-target export 100:2
Router(config-vrf)# route-target import 100:2
```

```
Router(config-vrf)# exit

Router(config)# ip cef
Router(config)# interface Loopback1
Router(config-if)# ip vrf forwarding v1
Router(config-if)# ip address 3.3.1.3 255.255.255.0
Router(config-if)# exit

Router(config)# interface Loopback2
Router(config-if)# ip vrf forwarding v2
Router(config-if)# ip address 3.3.2.3 255.255.255.0
Router(config-if)# exit

Router(config)# interface gigabitthernet1/1/0.10
Router(config-if)# encapsulation dot1q 10
Router(config-if)# ip vrf forwarding v1
Router(config-if)# ip address 38.0.0.3 255.255.255.0
Router(config-if)# exit

Router(config)# interface gigabitethernet1/1/0.20
Router(config-if)# encapsulation dot1q 20
Router(config-if)# ip vrf forwarding v2
Router(config-if)# ip address 83.0.0.3 255.255.255.0
Router(config-if)# exit

Router(config)# router bgp 100
Router(config-router)# address-family ipv4 vrf v2
Router(config-router-af)# neighbor 83.0.0.8 remote-as 800
Router(config-router-af)# neighbor 83.0.0.8 activate
Router(config-router-af)# network 3.3.2.0 mask 255.255.255.0
Router(config-router-af)# exit
Router(config-router)# address-family ipv4 vrf vl
Router(config-router-af)# neighbor 38.0.0.8 remote-as 800
Router(config-router-af)# neighbor 38.0.0.8 activate
Router(config-router-af)# network 3.3.1.0 mask 255.255.255.0
Router(config-router-af)# end
```

Displaying Multi-VRF CE Status

You can use the privileged EXEC commands in Table 32-13 to display information about multi-VRF CE configuration and status.

Table 32-13 Commands for Displaying Multi-VRF CE Information

Command	Purpose
show ip protocols vrf *vrf-name*	Display routing protocol information associated with a VRF.
show ip route vrf *vrf-name* [**connected**] [*protocol* [*as-number*]] [**list**] [**mobile**] [**odr**] [**profile**] [**static**] [**summary**] [**supernets-only**]	Display IP routing table information associated with a VRF.
show ip vrf [**brief** \| **detail** \| **interfaces**] [*vrf-name*]	Display information about the defined VRF instances.

For more information about the information in the displays, refer to the *Cisco IOS Switching Services Command Reference, Release 12.2*.

Configuring Protocol-Independent Features

This section describes how to configure IP routing protocol-independent features. For a complete description of the IP routing protocol-independent commands in this chapter, see the "IP Routing Protocol-Independent Commands" chapter of the *Cisco IOS IP Command Reference, Volume 2 of 3: Routing Protocols, Release 12.2.*

These sections contain this configuration information:

- Configuring Cisco Express Forwarding, page 32-69
- Configuring the Number of Equal-Cost Routing Paths, page 32-70
- Configuring Static Unicast Routes, page 32-71
- Specifying Default Routes and Networks, page 32-72
- Using Route Maps to Redistribute Routing Information, page 32-73
- Configuring Policy-Based Routing, page 32-76
- Filtering Routing Information, page 32-79
- Managing Authentication Keys, page 32-82

Configuring Cisco Express Forwarding

Cisco Express Forwarding (CEF) is a Layer 3 IP switching technology used to optimize network performance. CEF implements an advanced IP look-up and forwarding algorithm to deliver maximum Layer 3 switching performance. CEF is less CPU-intensive than fast switching route caching, allowing more CPU processing power to be dedicated to packet forwarding. In dynamic networks, fast switching cache entries are frequently invalidated because of routing changes, which can cause traffic to be process switched using the routing table, instead of fast switched using the route cache. CEF use the Forwarding Information Base (FIB) lookup table to perform destination-based switching of IP packets.

The two main components in CEF are the distributed FIB and the distributed adjacency tables.

- The FIB is similar to a routing table or information base and maintains a mirror image of the forwarding information in the IP routing table. When routing or topology changes occur in the network, the IP routing table is updated, and those changes are reflected in the FIB. The FIB maintains next-hop address information based on the information in the IP routing table. Because the FIB contains all known routes that exist in the routing table, CEF eliminates route cache maintenance, is more efficient for switching traffic, and is not affected by traffic patterns.
- Nodes in the network are said to be adjacent if they can reach each other with a single hop across a link layer. CEF uses adjacency tables to prepend Layer 2 addressing information. The adjacency table maintains Layer 2 next-hop addresses for all FIB entries.

Because the switch uses Application Specific Integrated Circuits (ASICs) to achieve Gigabit-speed line rate IP traffic, CEF forwarding applies only to the software-forwarding path, that is, traffic that is forwarded by the CPU.

CEF is enabled globally by default. If for some reason it is disabled, you can re-enable it by using the **ip cef** global configuration command.

The default configuration is CEF enabled on all Layer 3 interfaces. Entering the **no ip route-cache cef** interface configuration command disables CEF for traffic that is being forwarded by software. This command does not affect the hardware forwarding path. Disabling CEF and using the **debug ip packet**

detail privileged EXEC command can be useful to debug software-forwarded traffic. To enable CEF on an interface for the software-forwarding path, use the **ip route-cache cef** interface configuration command.

Caution Although the **no ip route-cache cef** interface configuration command to disable CEF on an interface is visible in the CLI, we strongly recommend that you do not disable CEF on interfaces except for debugging purposes.

Beginning in privileged EXEC mode, follow these steps to enable CEF globally and on an interface for software-forwarded traffic if it has been disabled:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip cef**	Enable CEF operation.
Step 3	**interface** *interface-id*	Enter interface configuration mode, and specify the Layer 3 interface to configure.
Step 4	**no shutdown**	Enable the interface if necessary. By default, UNIs are disabled and NNIs are enabled.
Step 5	**ip route-cache cef**	Enable CEF on the interface for software-forwarded traffic.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show ip cef**	Display the CEF status on all interfaces.
Step 8	**show cef linecard** [**detail**]	Display CEF-related interface information.
Step 9	**show cef interface** [*interface-id*]	Display detailed CEF information for all interfaces or the specified interface.
Step 10	**show adjacency**	Display CEF adjacency table information.
Step 11	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Configuring the Number of Equal-Cost Routing Paths

When a router has two or more routes to the same network with the same metrics, these routes can be thought of as having an equal cost. The term *parallel path* is another way to see occurrences of equal-cost routes in a routing table. If a router has two or more equal-cost paths to a network, it can use them concurrently. Parallel paths provide redundancy in case of a circuit failure and also enable a router to load balance packets over the available paths for more efficient use of available bandwidth.

Although the router automatically learns about and configures equal-cost routes, you can control the maximum number of parallel paths supported by an IP routing protocol in its routing table.

Beginning in privileged EXEC mode, follow these steps to change the maximum number of parallel paths installed in a routing table from the default:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**router** {**bgp** \| **rip** \| **ospf** \| **eigrp**}	Enter router configuration mode.

	Command	Purpose
Step 3	**maximum-paths** *maximum*	Set the maximum number of parallel paths for the protocol routing table. The range is from 1 to 8; the default is 4 for most IP routing protocols, but only 1 for BGP.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show ip protocols**	Verify the setting in the *Maximum path* field.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no maximum-paths** router configuration command to restore the default value.

Configuring Static Unicast Routes

Static unicast routes are user-defined routes that cause packets moving between a source and a destination to take a specified path. Static routes can be important if the router cannot build a route to a particular destination and are useful for specifying a gateway of last resort to which all unroutable packets are sent.

Beginning in privileged EXEC mode, follow these steps to configure a static route:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip route** *prefix mask* {*address* \| *interface*} [*distance*]	Establish a static route.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show ip route**	Display the current state of the routing table to verify the configuration.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no ip route** *prefix mask* {*address* | *interface*} global configuration command to remove a static route.

The switch retains static routes until you remove them. However, you can override static routes with dynamic routing information by assigning administrative distance values. Each dynamic routing protocol has a default administrative distance, as listed in Table 32-14. If you want a static route to be overridden by information from a dynamic routing protocol, set the administrative distance of the static route higher than that of the dynamic protocol.

Table 32-14 Dynamic Routing Protocol Default Administrative Distances

Route Source	Default Distance
Connected interface	0
Static route	1
Enhanced IRGP summary route	5
External BGP	20
Internal Enhanced IGRP	90
IGRP	100

Table 32-14 Dynamic Routing Protocol Default Administrative Distances (continued)

Route Source	Default Distance
OSPF	110
Internal BGP	200
Unknown	225

Static routes that point to an interface are advertised through RIP, IGRP, and other dynamic routing protocols, whether or not static **redistribute** router configuration commands were specified for those routing protocols. These static routes are advertised because static routes that point to an interface are considered in the routing table to be connected and hence lose their static nature. However, if you define a static route to an interface that is not one of the networks defined in a network command, no dynamic routing protocols advertise the route unless a **redistribute** static command is specified for these protocols.

When an interface goes down, all static routes through that interface are removed from the IP routing table. When the software can no longer find a valid next hop for the address specified as the forwarding router's address in a static route, the static route is also removed from the IP routing table.

Specifying Default Routes and Networks

A router might not be able to learn the routes to all other networks. To provide complete routing capability, you can use some routers as smart routers and give the remaining routers default routes to the smart router. (Smart routers have routing table information for the entire internetwork.) These default routes can be dynamically learned or can be configured in the individual routers. Most dynamic interior routing protocols include a mechanism for causing a smart router to generate dynamic default information that is then forwarded to other routers.

If a router has a directly connected interface to the specified default network, the dynamic routing protocols running on that device generate a default route. In RIP, it advertises the pseudonetwork 0.0.0.0.s

A router that is generating the default for a network also might need a default of its own. One way a router can generate its own default is to specify a static route to the network 0.0.0.0 through the appropriate device.

Beginning in privileged EXEC mode, follow these steps to define a static route to a network as the static default route:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip default-network** *network number*	Specify a default network.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show ip route**	Display the selected default route in the gateway of last resort display.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no ip default-network** *network number* global configuration command to remove the route.

When default information is passed through a dynamic routing protocol, no further configuration is required. The system periodically scans its routing table to choose the optimal default network as its default route. In IGRP networks, there might be several candidate networks for the system default. Cisco routers use administrative distance and metric information to set the default route or the gateway of last resort.

If dynamic default information is not being passed to the system, candidates for the default route are specified with the **ip default-network** global configuration command. If this network appears in the routing table from any source, it is flagged as a possible choice for the default route. If the router has no interface on the default network, but does have a path to it, the network is considered as a possible candidate, and the gateway to the best default path becomes the gateway of last resort.

Using Route Maps to Redistribute Routing Information

The switch can run multiple routing protocols simultaneously, and it can redistribute information from one routing protocol to another. Redistributing information from one routing protocol to another applies to all supported IP-based routing protocols.

You can also conditionally control the redistribution of routes between routing domains by defining enhanced packet filters or route maps between the two domains. The **match** and **set** route-map configuration commands define the condition portion of a route map. The **match** command specifies that a criterion must be matched; the **set** command specifies an action to be taken if the routing update meets the conditions defined by the match command. Although redistribution is a protocol-independent feature, some of the **match** and **set** route-map configuration commands are specific to a particular protocol.

One or more **match** commands and one or more **set** commands follow a **route-map** command. If there are no **match** commands, everything matches. If there are no **set** commands, nothing is done, other than the match. Therefore, you need at least one **match** or **set** command.

You can also identify route-map statements as **permit** or **deny**. If the statement is marked as a deny, the packets meeting the match criteria are sent back through the normal forwarding channels (destination-based routing). If the statement is marked as permit, set clauses are applied to packets meeting the match criteria. Packets that do not meet the match criteria are forwarded through the normal routing channel.



Note Although each of Steps 3 through 14 in the following section is optional, you must enter at least one **match** route-map configuration command and one **set** route-map configuration command.

Beginning in privileged EXEC mode, follow these steps to configure a route map for redistribution:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**route-map** *map-tag* [**permit** \| **deny**] [*sequence number*]	Define any route maps used to control redistribution and enter route-map configuration mode. *map-tag*—A meaningful name for the route map. The **redistribute** router configuration command uses this name to reference this route map. Multiple route maps might share the same map tag name. (Optional) If **permit** is specified and the match criteria are met for this route map, the route is redistributed as controlled by the set actions. If **deny** is specified, the route is not redistributed. *sequence number* (Optional)— Number that indicates the position a new route map is to have in the list of route maps already configured with the same name.
Step 3	**match as-path** *path-list-number*	Match a BGP AS path access list.
Step 4	**match community-list** *community-list-number* [**exact**]	Match a BGP community list.
Step 5	**match ip address** {*access-list-number* \| *access-list-name*} [*...access-list-number* \| *...access-list-name*]	Match a standard access list by specifying the name or number. It can be an integer from 1 to 199.
Step 6	**match metric** *metric-value*	Match the specified route metric. The *metric-valu*e can be an EIGRP metric with a specified value from 0 to 4294967295.
Step 7	**match ip next-hop** {*access-list-number* \| *access-list-name*} [*...access-list-number* \| *...access-list-name*]	Match a next-hop router address passed by one of the access lists specified (numbered from 1 to 199).
Step 8	**match tag** *tag value* [*...tag-value*]	Match the specified tag value in a list of one or more route tag values. Each can be an integer from 0 to 4294967295.
Step 9	**match interface** *type number* [*...type number*]	Match the specified next hop route out one of the specified interfaces.
Step 10	**match ip route-source** {*access-list-number* \| *access-list-name*} [*...access-list-number* \| *...access-list-name*]	Match the address specified by the specified advertised access lists.
Step 11	**match route-type** {**local** \| **internal** \| **external** [**type-1** \| **type-2**]}	Match the specified **route-type**: • **local**—Locally generated BGP routes. • **internal**—OSPF intra-area and interarea routes or EIGRP internal routes. • **external**—OSPF external routes (Type 1 or Type 2) or EIGRP external routes.
Step 12	**set dampening** *halflife reuse suppress max-suppress-time*	Set BGP route dampening factors.
Step 13	**set local-preference** *value*	Assign a value to a local BGP path.

	Command	Purpose
Step 14	**set origin** {**igp** \| **egp** *as* \| **incomplete**}	Set the BGP origin code.
Step 15	**set as-path** {**tag** \| **prepend** *as-path-string*}	Modify the BGP autonomous system path.
Step 16	**set level** {**level-1** \| **level-2** \| **level-1-2** \| **stub-area** \| **backbone**}	Set the level for routes that are advertised into the specified area of the routing domain. The **stub-area** and **backbone** are OSPF NSSA and backbone areas.
Step 17	**set metric** *metric value*	Set the metric value to give the redistributed routes (for EIGRP only). The *metric value* is an integer from -294967295 to 294967295.
Step 18	**set metric** *bandwidth delay reliability loading mtu*	Set the metric value to give the redistributed routes (for EIGRP only): • *bandwidth*—Metric value or IGRP bandwidth of the route in kilobits per second in the range 0 to 4294967295 • *delay*—Route delay in tens of microseconds in the range 0 to 4294967295. • *reliability*—Likelihood of successful packet transmission expressed as a number between 0 and 255, where 255 means 100 percent reliability and 0 means no reliability. • *loading*— Effective bandwidth of the route expressed as a number from 0 to 255 (255 is 100 percent loading). • *mtu*—Minimum maximum transmission unit (MTU) size of the route in bytes in the range 0 to 4294967295.
Step 19	**set metric-type** {**type-1** \| **type-2**}	Set the OSPF external metric type for redistributed routes.
Step 20	**set metric-type internal**	Set the multi-exit discriminator (MED) value on prefixes advertised to external BGP neighbor to match the IGP metric of the next hop.
Step 21	**set weight**	Set the BGP weight for the routing table. The value can be from 1 to 65535.
Step 22	**end**	Return to privileged EXEC mode.
Step 23	**show route-map**	Display all route maps configured or only the one specified to verify configuration.
Step 24	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To delete an entry, use the **no route-map** *map tag* global configuration command or the **no match** or **no set** route-map configuration commands.

You can distribute routes from one routing domain into another and control route distribution.

Beginning in privileged EXEC mode, follow these steps to control route redistribution. Note that the keywords are the same as defined in the previous procedure.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**router** {**bgp** \| **rip** \| **ospf** \| **eigrp**}	Enter router configuration mode.
Step 3	**redistribute** *protocol* [*process-id*] {**level-1** \| **level-1-2** \| **level-2**} [**metric** *metric-value*] [**metric-type** *type-value*] [**match internal** \| **external** *type-value*] [**tag** *tag-value*] [**route-map** *map-tag*] [**weight** *weight*] [**subnets**]	Redistribute routes from one routing protocol to another routing protocol. If no route-maps are specified, all routes are redistributed. If the keyword **route-map** is specified with no *map-tag*, no routes are distributed.
Step 4	**default-metric** *number*	Cause the current routing protocol to use the same metric value for all redistributed routes (BGP, RIP and OSPF).
Step 5	**default-metric** *bandwidth delay reliability loading mtu*	Cause the EIGRP routing protocol to use the same metric value for all non-EIGRP redistributed routes.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show route-map**	Display all route maps configured or only the one specified to verify configuration.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable redistribution, use the **no** form of the commands.

The metrics of one routing protocol do not necessarily translate into the metrics of another. In these situations, an artificial metric is assigned to the redistributed route. Uncontrolled exchanging of routing information between different routing protocols can create routing loops and seriously degrade network operation.

If you have not defined a default redistribution metric that replaces metric conversion, some automatic metric translations occur between routing protocols:

- RIP can automatically redistribute static routes. It assigns static routes a metric of 1 (directly connected).
- Any protocol can redistribute other routing protocols if a default mode is in effect.

Configuring Policy-Based Routing

You can use policy-based routing (PBR) to configure a defined policy for traffic flows. By using PBR, you can have more control over routing by reducing the reliance on routes derived from routing protocols. PBR can specify and implement routing policies that allow or deny paths based on:

- Identity of a particular end system
- Application
- Protocol

You can use PBR to provide equal-access and source-sensitive routing, routing based on interactive versus batch traffic, or routing based on dedicated links. For example, you could transfer stock records to a corporate office on a high-bandwidth, high-cost link for a short time while transmitting routine application data such as e-mail over a low-bandwidth, low-cost link.

With PBR, you classify traffic using access control lists (ACLs) and then make traffic go through a different path. PBR is applied to incoming packets. All packets received on an interface with PBR enabled are passed through route maps. Based on the criteria defined in the route maps, packets are forwarded (routed) to the appropriate next hop.

- If packets do not match any route map statements, all set clauses are applied.
- If a statement is marked as deny, packets meeting the match criteria are sent through normal forwarding channels, and destination-based routing is performed.
- If a statement is marked as permit and the packets do not match any route-map statements, the packets are sent through the normal forwarding channels, and destination-based routing is performed.

For more information about configuring route maps, see the "Using Route Maps to Redistribute Routing Information" section on page 32-73.

You can use standard IP ACLs to specify match criteria for a source address or extended IP ACLs to specify match criteria based on an application, a protocol type, or an end station. The process proceeds through the route map until a match is found. If no match is found, or if the route map is a deny, normal destination-based routing occurs. There is an implicit deny at the end of the list of match statements.

If match clauses are satisfied, you can use a set clause to specify the IP addresses identifying the next hop router in the path.

For details about PBR commands and keywords, see the *Cisco IOS IP Command Reference, Volume 2 of 3: Routing Protocols, Release 12.2*. For a list of PBR commands that are visible but not supported by the switch, see Appendix C, "Unsupported Commands in Cisco IOS Release 12.2(25)EX."

PBR Configuration Guidelines

Before configuring PBR, you should be aware of this information:

- Multicast traffic is not policy-routed. PBR applies to only to unicast traffic.
- You can enable PBR on a routed port or an SVI.
- You can apply a policy route map to an EtherChannel port channel in Layer 3 mode, but you cannot apply a policy route map to a physical interface that is a member of the EtherChannel. If you try to do so, the command is rejected. When a policy route map is applied to a physical interface, that interface cannot become a member of an EtherChannel.
- You can define a maximum of 246 IP policy route maps on the switch.
- You can define a maximum of 512 access control entries (ACEs) for PBR on the switch.
- To use PBR, you must first enable the routing template by using the **sdm prefer routing** global configuration command. PBR is not supported with the VLAN or default template. For more information on the SDM templates, see Chapter 6, "Configuring SDM Templates."
- VRF and PBR are mutually-exclusive on a switch interface. You cannot enable VRF when PBR is enabled on an interface. In contrast, you cannot enable PBR when VRF is enabled on an interface.
- The number of TCAM entries used by PBR depends on the route map itself, the ACLs used, and the order of the ACLs and route-map entries.
- Policy-based routing based on packet length, IP precedence and TOS, set interface, set default next hop, or set default interface are not supported. Policy maps with no valid set actions or with set action set to *Don't Fragment* are not supported.

Enabling PBR

By default, PBR is disabled on the switch. To enable PBR, you must create a route map that specifies the match criteria and the resulting action if all of the match clauses are met. Then, you must enable PBR for that route map on an interface. All packets arriving on the specified interface matching the match clauses are subject to PBR.

PBR can be fast-switched or implemented at speeds that do not slow down the switch. Fast-switched PBR supports most match and set commands. PBR must be enabled before you enable fast-switched PBR. Fast-switched PBR is disabled by default.

Packets that are generated by the switch, or local packets, are not normally policy-routed. When you globally enable local PBR on the switch, all packets that originate on the switch are subject to local PBR. Local PBR is disabled by default.

Beginning in privileged EXEC mode, follow these steps to configure PBR:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**route-map** *map-tag* [**permit** \| **deny**] [*sequence number*]	Define any route maps used to control where packets are output, and enter route-map configuration mode. *map-tag*—A meaningful name for the route map. The **ip policy route-map** interface configuration command uses this name to reference the route map. Multiple route maps might share the same map tag name. (Optional) If **permit** is specified and the match criteria are met for this route map, the route is policy-routed as controlled by the set actions. If **deny** is specified, the route is not policy-routed. *sequence number* (Optional)— Number that shows the position of a new route map in the list of route maps already configured with the same name.
Step 3	**match ip address** {*access-list-number* \| *access-list-name*} [*...access-list-number* \| *...access-list-name*]	Match the source and destination IP address that is permitted by one or more standard or extended access lists. If you do not specify a **match** command, the route map applies to all packets.
Step 4	**set ip next-hop** *ip-address* [*...ip-address*]	Specify the action to take on the packets that match the criteria. Set next hop to which to route the packet (the next hop must be adjacent).
Step 5	**exit**	Return to global configuration mode.
Step 6	**interface** *interface-id*	Enter interface configuration mode, and specify the interface to configure.
Step 7	**no shutdown**	Enable the interface if necessary. By default, UNIs are disabled and NNIs are enabled.

	Command	Purpose
Step 8	**ip policy route-map** *map-tag*	Enable PBR on a Layer 3 interface, and identify the route map to use. You can configure only one route map on an interface. However, you can have multiple route map entries with different sequence numbers. These entries are evaluated in sequence number order until the first match. If there is no match, packets are routed as usual.
Step 9	**ip route-cache policy**	(Optional) Enable fast-switching PBR. You must first enable PBR before enabling fast-switching PBR.
Step 10	**exit**	Return to global configuration mode.
Step 11	**ip local policy route-map** *map-tag*	(Optional) Enable local PBR to perform policy-based routing on packets originating at the switch. This applies to packets generated by the switch and not to incoming packets.
Step 12	**end**	Return to privileged EXEC mode.
Step 13	**show route-map** [*map-name*]	(Optional) Display all route maps configured or only the one specified to verify configuration.
Step 14	**show ip policy**	(Optional) Display policy route maps attached to interfaces.
Step 15	**show ip local policy**	(Optional) Display whether or not local policy routing is enabled and, if so, the route map being used.
Step 16	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no route-map** *map-tag* global configuration command or the **no match** or **no set** route-map configuration commands to delete an entry. Use the **no ip policy route-map** *map-tag* interface configuration command to disable PBR on an interface. Use the **no ip route-cache policy** interface configuration command to disable fast-switching PBR. Use the no **ip local policy route-map** *map-tag* global configuration command to disable policy-based routing on packets originating on the switch.

Filtering Routing Information

You can filter routing protocol information by performing the tasks described in this section.

Note When routes are redistributed between OSPF processes, no OSPF metrics are preserved.

Setting Passive Interfaces

To prevent other routers on a local network from dynamically learning about routes, you can use the **passive-interface** router configuration command to keep routing update messages from being sent through a router interface. When you use this command in the OSPF protocol, the interface address you specify as passive appears as a stub network in the OSPF domain. OSPF routing information is neither sent nor received through the specified router interface.

In networks with many interfaces, to avoid having to manually set them as passive, you can set all interfaces to be passive by default by using the **passive-interface default** router configuration command and manually setting interfaces where adjacencies are desired.

Beginning in privileged EXEC mode, follow these steps to configure passive interfaces:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**router** {**bgp** \| **rip** \| **ospf** \| **eigrp**}	Enter router configuration mode.
Step 3	**passive-interface** *interface-id*	Suppress sending routing updates through the specified Layer 3 interface.
Step 4	**passive-interface default**	(Optional) Set all interfaces as passive by default.
Step 5	**no passive-interface** *interface type*	(Optional) Activate only those interfaces that need to have adjacencies sent.
Step 6	**network** *network-address*	(Optional) Specify the list of networks for the routing process. The *network-address* is an IP address.
Step 7	**end**	Return to privileged EXEC mode.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use a network monitoring privileged EXEC command such as **show ip ospf interface** to verify the interfaces that you enabled as passive, or use the **show ip interface** privileged EXEC command to verify the interfaces that you enabled as active.

To re-enable the sending of routing updates, use the **no passive-interface** *interface-id* router configuration command. The **default** keyword sets all interfaces as passive by default. You can then configure individual interfaces where you want adjacencies by using the **no passive-interface** router configuration command. The **default** keyword is useful in Internet service provider and large enterprise networks where many of the distribution routers have more than 200 interfaces.

Controlling Advertising and Processing in Routing Updates

You can use the **distribute-list** router configuration command with access control lists to suppress routes from being advertised in routing updates and to prevent other routers from learning one or more routes. When used in OSPF, this feature applies to only external routes, and you cannot specify an interface name.

You can also use a **distribute-list** router configuration command to avoid processing certain routes listed in incoming updates. (This feature does not apply to OSPF.)

Beginning in privileged EXEC mode, follow these steps to control the advertising or processing of routing updates:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**router** {**bgp** \| **rip** \| **eigrp**}	Enter router configuration mode.
Step 3	**distribute-list** {*access-list-number* \| *access-list-name*} **out** [*interface-name* \| *routing process* \| *autonomous-system-number*]	Permit or deny routes from being advertised in routing updates, depending upon the action listed in the access list.
Step 4	**distribute-list** {*access-list-number* \| *access-list-name*} **in** [*type-number*]	Suppress processing in routes listed in updates.

	Command	Purpose
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no distribute-list in** router configuration command to change or cancel a filter. To cancel suppression of network advertisements in updates, use the **no distribute-list out** router configuration command.

Filtering Sources of Routing Information

Because some routing information might be more accurate than others, you can use filtering to prioritize information coming from different sources. An *administrative distance* is a rating of the trustworthiness of a routing information source, such as a router or group of routers. In a large network, some routing protocols can be more reliable than others. By specifying administrative distance values, you enable the router to intelligently discriminate between sources of routing information. The router always picks the route whose routing protocol has the lowest administrative distance. Table 32-14 on page 32-71 shows the default administrative distances for various routing information sources.

Because each network has its own requirements, there are no general guidelines for assigning administrative distances.

Beginning in privileged EXEC mode, follow these steps to filter sources of routing information:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**router** {**bgp** \| **rip** \| **ospf** \| **eigrp**}	Enter router configuration mode.
Step 3	**distance** *weight* {*ip-address* {*ip-address mask*}} [*ip access list*]	Define an administrative distance. *weight*—The administrative distance as an integer from 10 to 255. Used alone, *weight* specifies a default administrative distance that is used when no other specification exists for a routing information source. Routes with a distance of 255 are not installed in the routing table. (Optional) *ip access list*—An IP standard or extended access list to be applied to incoming routing updates.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show ip protocols**	Display the default administrative distance for a specified routing process.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove a distance definition, use the **no distance** router configuration command.

Managing Authentication Keys

Key management is a method of controlling authentication keys used by routing protocols. Not all protocols can use key management. Authentication keys are available for EIGRP and RIP Version 2.

Before you manage authentication keys, you must enable authentication. See the appropriate protocol section to see how to enable authentication for that protocol. To manage authentication keys, define a key chain, identify the keys that belong to the key chain, and specify how long each key is valid. Each key has its own key identifier (specified with the **key** *number* key chain configuration command), which is stored locally. The combination of the key identifier and the interface associated with the message uniquely identifies the authentication algorithm and Message Digest 5 (MD5) authentication key in use.

You can configure multiple keys with life times. Only one authentication packet is sent, regardless of how many valid keys exist. The software examines the key numbers in order from lowest to highest, and uses the first valid key it encounters. The lifetimes allow for overlap during key changes. Note that the router must know these lifetimes.

Beginning in privileged EXEC mode, follow these steps to manage authentication keys:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**key chain** *name-of-chain*	Identify a key chain, and enter key chain configuration mode.
Step 3	**key** *number*	Identify the key number. The range is 0 to 2147483647.
Step 4	**key-string** *text*	Identify the key string. The string can contain from 1 to 80 uppercase and lowercase alphanumeric characters, but the first character cannot be a number.
Step 5	**accept-lifetime** *start-time* {**infinite** \| *end-time* \| **duration** *seconds*}	(Optional) Specify the time period during which the key can be received. The *start-time* and *end-time* syntax can be either *hh*:*mm*:*ss Month date year* or *hh*:*mm*:*ss date Month year*. The default is forever with the default *start-time* and the earliest acceptable date as January 1, 1993. The default *end-time* and **duration** is **infinite**.
Step 6	**send-lifetime** *start-time* {**infinite** \| *end-time* \| **duration** *seconds*}	(Optional) Specify the time period during which the key can be sent. The *start-time* and *end-time* syntax can be either *hh*:*mm*:*ss Month date year* or *hh*:*mm*:*ss date Month year*. The default is forever with the default *start-time* and the earliest acceptable date as January 1, 1993. The default *end-time* and **duration** is infinite.
Step 7	**end**	Return to privileged EXEC mode.
Step 8	**show key chain**	Display authentication key information.
Step 9	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the key chain, use the **no key chain** *name-of-chain* global configuration command.

Monitoring and Maintaining the IP Network

You can remove all contents of a particular cache, table, or database. You can also display specific statistics. Use the privileged EXEC commands in Table 32-15 to clear routes or display status:

Table 32-15 Commands to Clear IP Routes or Display Route Status

Command	Purpose
clear ip route {*network* [*mask* \| *]}	Clear one or more routes from the IP routing table.
show ip protocols	Display the parameters and state of the active routing protocol process.
show ip route [*address* [*mask*] [**longer-prefixes**]] \| [*protocol* [*process-id*]]	Display the current state of the routing table.
show ip route summary	Display the current state of the routing table in summary form.
show ip route supernets-only	Display supernets.
show ip cache	Display the routing table used to switch IP traffic.
show route-map [*map-name*]	Display all route maps configured or only the one specified.



CHAPTER 33

Configuring HSRP

This chapter describes how to use Hot Standby Router Protocol (HSRP) on the Cisco ME 3400 Ethernet Access switch to provide routing redundancy for routing IP traffic without being dependent on the availability of any single router. The switch must be running the metro IP access image to support HSRP.

Note For complete syntax and usage information for the commands used in this chapter, see the switch command reference for this release and the *Cisco IOS IP Command Reference, Volume 1 of 3: Addressing and Services, Release 12.2*.

This chapter consists of these sections:

- Understanding HSRP, page 33-1
- Configuring HSRP, page 33-4
- Displaying HSRP Configurations, page 33-11

Understanding HSRP

HSRP is Cisco's standard method of providing high network availability by providing first-hop redundancy for IP hosts on an IEEE 802 LAN configured with a default gateway IP address. HSRP routes IP traffic without relying on the availability of any single router. It enables a set of router interfaces to work together to present the appearance of a single virtual router or default gateway to the hosts on a LAN. When HSRP is configured on a network or segment, it provides a virtual Media Access Control (MAC) address and an IP address that is shared among a group of configured routers. HSRP allows two or more HSRP-configured routers to use the MAC address and IP network address of a virtual router. The virtual router does not exist; it represents the common target for routers that are configured to provide backup to each other. One of the routers is selected to be the active router and another to be the standby router, which assumes control of the group MAC address and IP address should the designated active router fail.

Note Routers in an HSRP group can be any router interface that supports HSRP, including routed ports and switch virtual interfaces (SVIs) on the switch.

HSRP provides high network availability by providing redundancy for IP traffic from hosts on networks. In a group of router interfaces, the active router is the router of choice for routing packets; the standby router is the router that takes over the routing duties when an active router fails or when preset conditions are met.

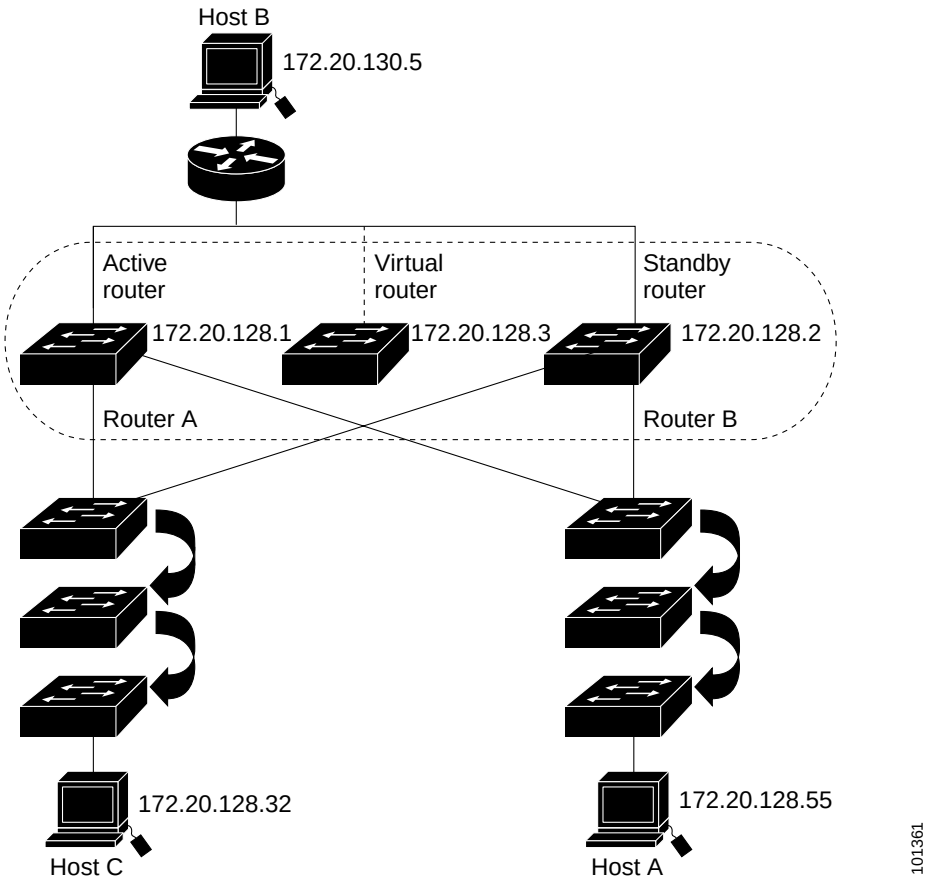
HSRP is useful for hosts that do not support a router discovery protocol and cannot switch to a new router when their selected router reloads or loses power. When HSRP is configured on a network segment, it provides a virtual MAC address and an IP address that is shared among router interfaces in a group of router interfaces running HSRP. The router selected by the protocol to be the active router receives and routes packets destined for the group's MAC address. For *n* routers running HSRP, there are *n +1* IP and MAC addresses assigned.

HSRP detects when the designated active router fails, and a selected standby router assumes control of the Hot Standby group's MAC and IP addresses. A new standby router is also selected at that time. Devices running HSRP send and receive multicast UDP-based hello packets to detect router failure and to designate active and standby routers. When HSRP is configured on an interface, Internet Control Message Protocol (ICMP) redirect messages are disabled by default for the interface.

You can configure multiple Hot Standby groups among switches that are operating in Layer 3 to make more use of the redundant routers. To do so, specify a group number for each Hot Standby command group you configure for an interface. For example, you might configure an interface on switch 1 as an active router and one on switch 2 as a standby router and also configure another interface on switch 2 as an active router with another interface on switch 1 as its standby router.

Figure 33-1 shows a segment of a network configured for HSRP. Each router is configured with the MAC address and IP network address of the virtual router. Instead of configuring hosts on the network with the IP address of Router A, you configure them with the IP address of the virtual router as their default router. When Host C sends packets to Host B, it sends them to the MAC address of the virtual router. If for any reason, Router A stops transferring packets, Router B responds to the virtual IP address and virtual MAC address and becomes the active router, assuming the active router duties. Host C continues to use the IP address of the virtual router to address packets destined for Host B, which Router B now receives and sends to Host B. Until Router A resumes operation, HSRP allows Router B to provide uninterrupted service to users on Host C's segment that need to communicate with users on Host B's segment and also continues to perform its normal function of handling packets between the Host A segment and Host B.

Figure 33-1 Typical HSRP Configuration



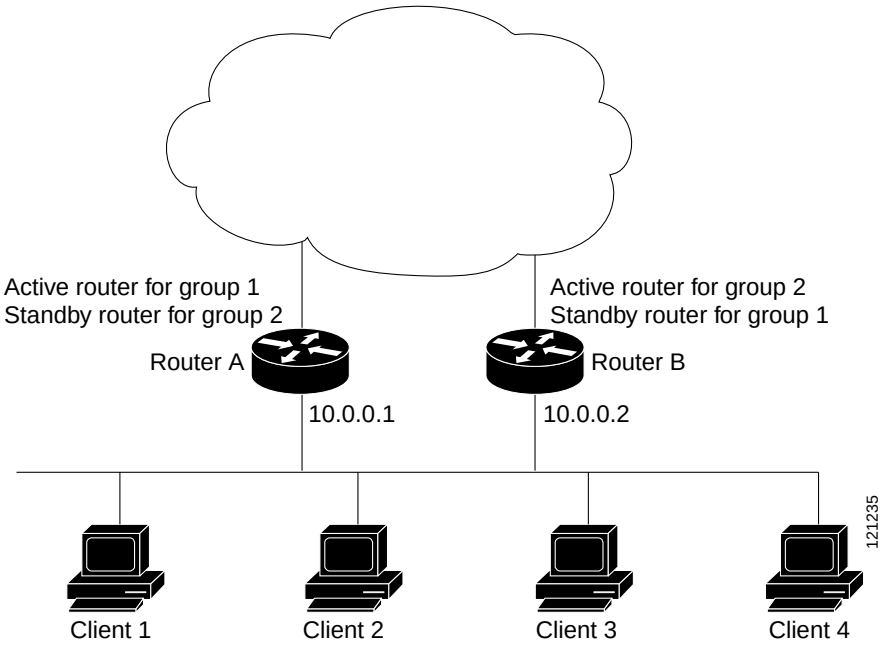
Multiple HSRP

The switch supports Multiple HSRP (MHSRP), an extension of HSRP that allows load sharing between two or more HSRP groups. You can configure MHSRP to achieve load balancing and to use two or more standby groups (and paths) from a host network to a server network. In Figure 33-2, half of the clients are configured for Router A, and half of the clients are configured for Router B. Together, the configuration for Routers A and B establish two HSRP groups. For group 1, Router A is the default active router because it has the assigned highest priority, and Router B is the standby router. For group 2, Router B is the default active router because it has the assigned highest priority, and Router A is the standby router. During normal operation, the two routers share the IP traffic load. When either router becomes unavailable, the other router becomes active and assumes the packet-transfer functions of the router that is unavailable.

See the "Configuring MHSRP" section on page 33-9 for the example configuration steps.

Note For MHSRP, you need to enter the **standby preempt** interface configuration command on the HSRP interfaces so that if a router fails and then comes back up, preemption occurs and restores load sharing

Figure 33-2 MHSRP Load Sharing



Configuring HSRP

These sections contain this configuration information:

- Default HSRP Configuration, page 33-4
- HSRP Configuration Guidelines, page 33-5
- Enabling HSRP, page 33-5
- Configuring HSRP Priority, page 33-6
- Configuring MHSRP, page 33-9
- Configuring HSRP Authentication and Timers, page 33-9
- Enabling HSRP Support for ICMP Redirect Messages, page 33-11

Default HSRP Configuration

Table 33-1 shows the default HSRP configuration.

Table 33-1 Default HSRP Configuration

Feature	Default Setting
HSRP groups	None configured
Standby group number	0
Standby MAC address	System assigned as: 0000.0c07.acXX, where *XX* is the HSRP group number
Standby priority	100

Table 33-1 Default HSRP Configuration (continued)

Feature	Default Setting
Standby delay	0 (no delay)
Standby track interface priority	10
Standby hello time	3 seconds
Standby holdtime	10 seconds

HSRP Configuration Guidelines

Follow these guidelines when configuring HSRP:

- HSRP can be configured on a maximum of 32 VLAN or routing interfaces.
- In the procedures, the specified interface must be one of these Layer 3 interfaces:
 - Routed port: a physical port configured as a Layer 3 port by entering the **no switchport** interface configuration command.
 - SVI: a VLAN interface created by using the **interface vlan** *vlan_id* global configuration command and by default a Layer 3 interface.
 - Etherchannel port channel in Layer 3 mode: a port-channel logical interface created by using the **interface port-channel** *port-channel-number* global configuration command and binding the Ethernet interface into the channel group. For more information, see the "Configuring Layer 3 EtherChannels" section on page 31-13.
- All Layer 3 interfaces must have IP addresses assigned to them. See the "Configuring Layer 3 Interfaces" section on page 9-20.

Enabling HSRP

The **standby ip** interface configuration command activates HSRP on the configured interface. If an IP address is specified, that address is used as the designated address for the Hot Standby group. If no IP address is specified, the address is learned through the standby function. You must configure at least one routing port on the cable with the designated address. Configuring an IP address always overrides another designated address currently in use.

When the **standby ip** command is enabled on an interface and proxy ARP is enabled, if the interface's Hot Standby state is active, proxy ARP requests are answered using the Hot Standby group MAC address. If the interface is in a different state, proxy ARP responses are suppressed.

Beginning in privileged EXEC mode, follow these steps to create or enable HSRP on a Layer 3 interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and enter the Layer 3 interface on which you want to enable HSRP.
Step 3	**no shutdown**	Enable the port, if necessary. By default, user network interfaces (UNIs) are disabled, and network node interfaces (NNIs) are enabled.

	Command	Purpose
Step 4	**no switchport**	If necessary, disable Layer 2 switching on the port to enable the Layer 3 interface.
Step 5	**standby** [*group-number*] **ip** [*ip-address* [**secondary**]]	Create (or enable) the HSRP group using its number and virtual IP address. • (Optional) *group-number*—The group number on the interface for which HSRP is being enabled. The range is 0 to 255; the default is 0. If there is only one HSRP group, you do not need to enter a group number. • (Optional on all but one interface) *ip-address*—The virtual IP address of the hot standby router interface. You must enter the virtual IP address for at least one of the interfaces; it can be learned on the other interfaces. • (Optional) **secondary**—The IP address is a secondary hot standby router interface. If neither router is designated as a secondary or standby router and no priorities are set, the primary IP addresses are compared and the higher IP address is the active router, with the next highest as the standby router.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show standby** [*interface-id* [*group*]]	Verify the configuration.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no standby** [*group-number*] **ip** [*ip-address*] interface configuration command to disable HSRP.

This example shows how to activate HSRP for group 1 on a port. The IP address used by the hot standby group is learned by using HSRP.

Note This procedure is the minimum number of steps required to enable HSRP. Other configuration is optional.

```
Switch# configure terminal
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# no switchport
Switch(config-if)# standby 1 ip
Switch(config-if)# end
Switch# show standby
```

Configuring HSRP Priority

The **standby priority**, **standby preempt**, and **standby track** interface configuration commands are all used to set characteristics for finding active and standby routers and behavior regarding when a new active router takes over.

When configuring HSRP priority, follow these guidelines:

- Assigning priority helps select the active and standby routers. If preemption is enabled, the router with the highest priority becomes the designated active router. If priorities are equal, the primary IP addresses are compared, and the higher IP address has priority.

- The highest number (1 to 255) represents the highest priority (most likely to become the active router).
- When setting the priority, preempt, or both, you must specify at least one keyword (**priority**, **preempt**, or both).
- The priority of the device can change dynamically if an interface is configured with the **standby track** command and another interface on the router goes down.
- The **standby track** interface configuration command ties the router hot standby priority to the availability of its interfaces and is useful for tracking interfaces that are not configured for HSRP. When a tracked interface fails, the hot standby priority on the device on which tracking has been configured decreases by 10. If an interface is not tracked, its state changes do not affect the hot standby priority of the configured device. For each interface configured for hot standby, you can configure a separate list of interfaces to be tracked.
- The **standby track** *interface-priority* interface configuration command specifies how much to decrement the hot standby priority when a tracked interface goes down. When the interface comes back up, the priority is incremented by the same amount.
- When multiple tracked interfaces are down and *interface-priority* values have been configured, the configured priority decrements are cumulative. If tracked interfaces that were not configured with priority values fail, the default decrement is 10, and it is noncumulative.
- When routing is first enabled for the interface, it does not have a complete routing table. If it is configured to preempt, it becomes the active router, even though it is unable to provide adequate routing services. To solve this problem, configure a delay time to allow the router to update its routing table.

Beginning in privileged EXEC mode, use one or more of these steps to configure HSRP priority characteristics on an interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and enter the HSRP interface on which you want to set priority.
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled, and NNIs are enabled.
Step 4	**standby** [*group-number*] **priority** *priority* [**preempt** [**delay** *delay*]]	Set a **priority** value used in choosing the active router. The range is 1 to 255; the default priority is 100. The highest number represents the highest priority. • (Optional) *group-number*—The group number to which the command applies. • (Optional) **preempt—**Select so that when the local router has a higher priority than the active router, it assumes control as the active router. • (Optional) **delay**—Set to cause the local router to postpone taking over the active role for the shown number of seconds. The range is 0 to 3600(1 hour); the default is 0 (no delay before taking over). Use the **no** form of the command to restore the default values.

	Command	Purpose
Step 5	**standby** [*group-number*] [**priority** *priority*] **preempt** [**delay** *delay*]	Configure the router to **preempt**, which means that when the local router has a higher priority than the active router, it assumes control as the active router. • (Optional) *group-number*—The group number to which the command applies. • (Optional) **priority**—Enter to set or change the group priority. The range is 1 to 255; the default is 100. • (Optional) **delay**—Set to cause the local router to postpone taking over the active role for the number of seconds shown. The range is 0 to 3600 (1 hour); the default is 0 (no delay before taking over). Use the **no** form of the command to restore the default values.
Step 6	**standby** [*group-number*] **track** *type number* [*interface-priority*]	Configure an interface to track other interfaces so that if one of the other interfaces goes down, the device's Hot Standby priority is lowered. • (Optional) *group-number*—The group number to which the command applies. • *type*—Enter the interface type (combined with interface number) that is tracked. • *number*—Enter the interface number (combined with interface type) that is tracked. • (Optional) *interface-priority*—Enter the amount by which the hot standby priority for the router is decremented or incremented when the interface goes down or comes back up. The default value is 10.
Step 7	**end**	Return to privileged EXEC mode.
Step 8	**show running-config**	Verify the configuration of the standby groups.
Step 9	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no standby** [*group-number*] **priority** *priority* [**preempt** [**delay** *delay*]] and **no standby** [*group-number*] [**priority** *priority*] **preempt** [**delay** *delay*] interface configuration commands to restore default priority, preempt, and delay values.

Use the **no standby** [*group-number*] **track** *type number* [*interface-priority*] interface configuration command to remove the tracking.

This example activates a port, sets an IP address and a priority of 120 (higher than the default value), and waits for 300 seconds (5 minutes) before attempting to become the active router:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# no switchport
Switch(config-if)# standby ip 172.20.128.3
Switch(config-if)# standby priority 120 preempt delay 300
Switch(config-if)# end
```

Configuring MHSRP

To enable MHSRP and load balancing, you configure two routers as active routers for their groups, with virtual routers as standby routers. This example shows how to enable the MHSRP configuration shown in Figure 33-2. You need to enter the **standby preempt** interface configuration command on each HSRP interface so that if a router fails and comes back up, the preemption occurs and restores load balancing.

Router A is configured as the active router for group 1, and Router B is configured as the active router for group 2. The HSRP interface for Router A has an IP address of 10.0.0.1 with a group 1 standby priority of 110 (the default is 100). The HSRP interface for Router B has an IP address of 10.0.0.2 with a group 2 standby priority of 110.

Group 1 uses a virtual IP address of 10.0.0.3 and group 2 uses a virtual IP address of 10.0.0.4.

Router A Configuration

```
Switch# configure terminal
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# no switchport
Switch(config-if)# ip address 10.0.0.1 255.255.255.0
Switch(config-if)# standby 1 ip 10.0.0.3
Switch(config-if)# standby 1 priority 110
Switch(config-if)# standby 1 preempt
Switch(config-if)# standby 2 ip 10.0.0.4
Switch(config-if)# standby 2 preempt
Switch(config-if)# end
```

Router B Configuration

```
Switch# configure terminal
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# no switchport
Switch(config-if)# ip address 10.0.0.2 255.255.255.0
Switch(config-if)# standby 1 ip 10.0.0.3
Switch(config-if)# standby 1 preempt
Switch(config-if)# standby 2 ip 10.0.0.4
Switch(config-if)# standby 2 priority 110
Switch(config-if)# standby 2 preempt
Switch(config-if)# end
```

Configuring HSRP Authentication and Timers

You can optionally configure an HSRP authentication string or change the hello-time interval and holdtime.

When configuring these attributes, follow these guidelines:

- The authentication string is sent unencrypted in all HSRP messages. You must configure the same authentication string on all routers and access servers on a cable to ensure interoperation. Authentication mismatch prevents a device from learning the designated Hot Standby IP address and timer values from other routers configured with HSRP.
- Routers or access servers on which standby timer values are not configured can learn timer values from the active or standby router. The timers configured on an active router always override any other timer settings.
- All routers in a Hot Standby group should use the same timer values. Normally, the *holdtime* is greater than or equal to 3 times the *hellotime*.

Beginning in privileged EXEC mode, use one or more of these steps to configure HSRP authentication and timers on an interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and enter the HSRP interface on which you want to set authentication.
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled, and NNIs are enabled.
Step 4	**standby** [*group-number*] **authentication** *string*	(Optional) **authentication** *string*—Enter a string to be carried in all HSRP messages. The authentication string can be up to eight characters in length; the default string is **cisco.** (Optional) *group-number*—The group number to which the command applies.
Step 5	**standby** [*group-number*] **timers** *hellotime holdtime*	(Optional) Configure the time between hello packets and the time before other routers declare the active router to be down. • *group-number*—The group number to which the command applies. • *hellotime*—The hello interval in seconds. The range is from 1 to 255; the default is 3 seconds. • *holdtime*—The time in seconds before the active or standby router is declared to be down. The range is from 1 to 255; the default is 10 seconds.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show running-config**	Verify the configuration of the standby groups.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no standby** [*group-number*] **authentication** *string* interface configuration command to delete an authentication string. Use the **no standby** [*group-number*] **timers** *hellotime holdtime* interface configuration command to restore timers to their default values.

This example shows how to configure *word* as the authentication string required to allow Hot Standby routers in group 1 to interoperate:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# no switchport
Switch(config-if)# standby 1 authentication word
Switch(config-if)# end
```

This example shows how to set the timers on standby group 1 with the time between hello packets at 5 seconds and the time after which a router is considered down to be 15 seconds:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# no switchport
Switch(config-if)# standby 1 ip
Switch(config-if)# standby 1 timers 5 15
Switch(config-if)# end
```

Enabling HSRP Support for ICMP Redirect Messages

ICMP (Internet Control Message Protocol) redirect messages are automatically enabled on interfaces configured with HSRP. This feature filters outgoing ICMP redirect messages through HSRP, in which the next hop IP address might be changed to an HSRP virtual IP address. ICMP is a network layer Internet protocol that provides message packets to report errors and other information relevant to IP processing. ICMP provides diagnostic functions, such as sending and directing error packets to the host.

When the switch is running HSRP, make sure hosts do not discover the interface (or real) MAC addresses of routers in the HSRP group. If a host is redirected by ICMP to the real MAC address of a router and that router later fails, packets from the host are lost.

For more information, see the *Cisco IOS IP Configuration Guide, Release 12.2*.

Displaying HSRP Configurations

From privileged EXEC mode, use this command to display HSRP settings:

show standby [*interface-id* [*group*]] [**brief**] [**detail**]

You can display HSRP information for the whole switch, for a specific interface, for an HSRP group, or for an HSRP group on an interface. You can also specify whether to display a concise overview of HSRP information or detailed HSRP information. The default display is **detail**. If there are a large number of HSRP groups, using the **show standby** command without qualifiers can result in an unwieldy display.

This is a an example of output from the **show standby** privileged EXEC command, displaying HSRP information for two standby groups (group 1 and group 100):

```
Switch# show standby
VLAN1 - Group 1
   Local state is Standby, priority 105, may preempt
   Hellotime 3 holdtime 10
   Next hello sent in 00:00:02.182
   Hot standby IP address is 172.20.128.3 configured
   Active router is 172.20.128.1 expires in 00:00:09
   Standby router is local
   Standby virtual mac address is 0000.0c07.ac01
   Name is bbb
VLAN1 - Group 100
   Local state is Active, priority 105, may preempt
   Hellotime 3 holdtime 10
   Next hello sent in 00:00:02.262
   Hot standby IP address is 172.20.138.51 configured
   Active router is local
   Standby router is unknown expired
   Standby virtual mac address is 0000.0c07.ac64
   Name is test
```



CHAPTER 34

Configuring IP Multicast Routing

This chapter describes how to configure IP multicast routing on the Cisco ME 3400 Ethernet Access switch. IP multicasting is a more efficient way to use network resources, especially for bandwidth-intensive services such as audio and video. IP multicast routing enables a host (source) to send packets to a group of hosts (receivers) anywhere within the IP network by using a special form of IP address called the IP *multicast group address.* The sending host inserts the multicast group address into the IP destination address field of the packet, and IP multicast routers and multilayer switches forward incoming IP multicast packets out all interfaces that lead to members of the multicast group. Any host, regardless of whether it is a member of a group, can sent to a group. However, only the members of a group receive the message.

To use this feature, the switch must be running the metro IP access image.

Note For complete syntax and usage information for the commands used in this chapter, see the *Cisco IOS IP Command Reference, Volume 3 of 3: Multicast, Release 12.2.*

This chapter consists of these sections:

- Understanding Cisco's Implementation of IP Multicast Routing, page 34-2
- Configuring IP Multicast Routing, page 34-7
- Configuring Advanced PIM Features, page 34-21
- Configuring Optional IGMP Features, page 34-25
- Configuring Optional Multicast Routing Features, page 34-31
- Monitoring and Maintaining IP Multicast Routing, page 34-34

For information on configuring the Multicast Source Discovery Protocol (MSDP), see Chapter 35, "Configuring MSDP."

Understanding Cisco's Implementation of IP Multicast Routing

The switch supports these protocols to implement IP multicast routing:

- Internet Group Management Protocol (IGMP) is used among hosts on a LAN and the routers (and multilayer switches) on that LAN to track the multicast groups of which hosts are members.
- Protocol-Independent Multicast (PIM) protocol is used among routers and multilayer switches to track which multicast packets to forward to each other and to their directly connected LANs.

This section contains this information:

- Understanding IGMP, page 34-2
- Understanding PIM, page 34-3
- Configuring IP Multicast Routing, page 34-7

Understanding IGMP

To participate in IP multicasting, multicast hosts, routers, and multilayer switches must have the IGMP operating. This protocol defines the querier and host roles:

- A querier is a network device that sends query messages to discover which network devices are members of a given multicast group.
- A host is a receiver that sends report messages (in response to query messages) to inform a querier of a host membership.

A set of queriers and hosts that receive multicast data streams from the same source is called a multicast group. Queriers and hosts use IGMP messages to join and leave multicast groups.

Any host, regardless of whether it is a member of a group, can send to a group. However, only the members of a group receive the message. Membership in a multicast group is dynamic; hosts can join and leave at any time. There is no restriction on the location or number of members in a multicast group. A host can be a member of more than one multicast group at a time. How active a multicast group is and what members it has can vary from group to group and from time to time. A multicast group can be active for a long time, or it can be very short-lived. Membership in a group can constantly change. A group that has members can have no activity.

IP multicast traffic uses group addresses, which are class D addresses. The high-order bits of a Class D address are 1110. Therefore, host group addresses can be in the range 224.0.0.0 through 239.255.255.255. Multicast addresses in the range 224.0.0.0 to 24.0.0.255 are reserved for use by routing protocols and other network control traffic. The address 224.0.0.0 is guaranteed not to be assigned to any group.

IGMP packets are sent using these IP multicast group addresses:

- IGMP general queries are destined to the address 224.0.0.1 (all systems on a subnet).
- IGMP group-specific queries are destined to the group IP address for which the switch is querying.
- IGMP group membership reports are destined to the group IP address for which the switch is reporting.
- IGMP Version 2 (IGMPv2) leave messages are destined to the address 224.0.0.2 (all-multicast-routers on a subnet). In some old host IP stacks, leave messages might be destined to the group IP address rather than to the all-routers address.

IGMP Version 1

IGMP Version 1 (IGMPv1) primarily uses a query-response model that enables the multicast router and multilayer switch to find which multicast groups are active (have one or more hosts interested in a multicast group) on the local subnet. IGMPv1 has other processes that enable a host to join and leave a multicast group. For more information, see RFC 1112.

IGMP Version 2

IGMPv2 extends IGMP functionality by providing such features as the IGMP leave process to reduce leave latency, group-specific queries, and an explicit maximum query response time. IGMPv2 also adds the capability for routers to elect the IGMP querier without depending on the multicast protocol to perform this task. For more information, see RFC 2236.

Understanding PIM

PIM is called *protocol-independent*: regardless of the unicast routing protocols used to populate the unicast routing table, PIM uses this information to perform multicast forwarding instead of maintaining a separate multicast routing table.

PIM is defined in RFC 2362, *Protocol-Independent Multicast-Sparse Mode (PIM-SM): Protocol Specification*. PIM is defined in these Internet Engineering Task Force (IETF) Internet drafts:

- *Protocol Independent Multicast (PIM): Motivation and Architecture*
- *Protocol Independent Multicast (PIM), Dense Mode Protocol Specification*
- *Protocol Independent Multicast (PIM), Sparse Mode Protocol Specification*
- *draft-ietf-idmr-igmp-v2-06.txt, Internet Group Management Protocol, Version 2*
- *draft-ietf-pim-v2-dm-03.txt, PIM Version 2 Dense Mode*

PIM Versions

PIMv2 includes these improvements over PIMv1:

- A single, active rendezvous point (RP) exists per multicast group, with multiple backup RPs. This single RP compares to multiple active RPs for the same group in PIMv1.
- A bootstrap router (BSR) provides a fault-tolerant, automated RP discovery and distribution mechanism that enables routers and multilayer switches to dynamically learn the group-to-RP mappings.
- Sparse mode and dense mode are properties of a group, as opposed to an interface. We strongly recommend sparse-dense mode, as opposed to either sparse mode or dense mode only.
- PIM join and prune messages have more flexible encoding for multiple address families.
- A more flexible hello packet format replaces the query packet to encode current and future capability options.
- Register messages to an RP specify whether they are sent by a border router or a designated router.
- PIM packets are no longer inside IGMP packets; they are standalone packets.

PIM Modes

PIM can operate in dense mode (DM), sparse mode (SM), or in sparse-dense mode (PIM DM-SM), which handles both sparse groups and dense groups at the same time.

PIM DM

PIM DM builds source-based multicast distribution trees. In dense mode, a PIM DM router or multilayer switch assumes that all other routers or multilayer switches forward multicast packets for a group. If a PIM DM device receives a multicast packet and has no directly connected members or PIM neighbors present, a prune message is sent back to the source to stop unwanted multicast traffic. Subsequent multicast packets are not flooded to this router or switch on this pruned branch because branches without receivers are pruned from the distribution tree, leaving only branches that contain receivers.

When a new receiver on a previously pruned branch of the tree joins a multicast group, the PIM DM device detects the new receiver and immediately sends a graft message up the distribution tree toward the source. When the upstream PIM DM device receives the graft message, it immediately puts the interface on which the graft was received into the forwarding state so that the multicast traffic begins flowing to the receiver.

PIM SM

PIM SM uses shared trees and shortest-path-trees (SPTs) to distribute multicast traffic to multicast receivers in the network. In PIM SM, a router or multilayer switch assumes that other routers or switches do not forward multicast packets for a group, unless there is an explicit request for the traffic (join message). When a host joins a multicast group using IGMP, its directly connected PIM SM device sends PIM join messages toward the root, also known as the RP. This join message travels router-by-router toward the root, constructing a branch of the shared tree as it goes.

The RP keeps track of multicast receivers. It also registers sources through register messages received from the source's first-hop router (*designated router* [DR]) to complete the shared tree path from the source to the receiver. When using a shared tree, sources must send their traffic to the RP so that the traffic reaches all receivers.

Prune messages are sent up the distribution tree to prune multicast group traffic. This action permits branches of the shared tree or SPT that were created with explicit join messages to be torn down when they are no longer needed.

Auto-RP

This proprietary feature eliminates the need to manually configure the RP information in every router and multilayer switch in the network. For Auto-RP to work, you configure a Cisco router or multilayer switch as the mapping agent. It uses IP multicast to learn which routers or switches in the network are possible candidate RPs to receive candidate RP announcements. Candidate RPs periodically send multicast RP-announce messages to a particular group or group range to announce their availability.

Mapping agents listen to these candidate RP announcements and use the information to create entries in their Group-to-RP mapping caches. Only one mapping cache entry is created for any Group-to-RP range received, even if multiple candidate RPs are sending RP announcements for the same range. As the RP-announce messages arrive, the mapping agent selects the router or switch with the highest IP address as the active RP and stores this RP address in the Group-to-RP mapping cache.

Mapping agents periodically multicast the contents of their Group-to-RP mapping cache. Thus, all routers and switches automatically discover which RP to use for the groups they support. If a router or switch fails to receive RP-discovery messages and the Group-to-RP mapping information expires, it

switches to a statically configured RP that was defined with the **ip pim rp-address** global configuration command. If no statically configured RP exists, the router or switch changes the group to dense-mode operation.

Multiple RPs serve different group ranges or serve as hot backups of each other.

Bootstrap Router

PIMv2 BSR is another method to distribute group-to-RP mapping information to all PIM routers and multilayer switches in the network. It eliminates the need to manually configure RP information in every router and switch in the network. However, instead of using IP multicast to distribute group-to-RP mapping information, BSR uses hop-by-hop flooding of special BSR messages to distribute the mapping information.

The BSR is elected from a set of candidate routers and switches in the domain that have been configured to function as BSRs. The election mechanism is similar to the root-bridge election mechanism used in bridged LANs. The BSR election is based on the BSR priority of the device contained in the BSR messages that are sent hop-by-hop through the network. Each BSR device examines the message and forwards out all interfaces only the message that has either a higher BSR priority than its BSR priority or the same BSR priority, but with a higher BSR IP address. Using this method, the BSR is elected.

The elected BSR sends BSR messages with a TTL of 1. Neighboring PIMv2 routers or multilayer switches receive the BSR message and multicast it out all other interfaces (except the one on which it was received) with a TTL of 1. In this way, BSR messages travel hop-by-hop throughout the PIM domain. Because BSR messages contain the IP address of the current BSR, the flooding mechanism enables candidate RPs to automatically learn which device is the elected BSR.

Candidate RPs send candidate RP advertisements showing the group range for which they are responsible to the BSR, which stores this information in its local candidate-RP cache. The BSR periodically advertises the contents of this cache in BSR messages to all other PIM devices in the domain. These messages travel hop-by-hop through the network to all routers and switches, which store the RP information in the BSR message in their local RP cache. The routers and switches select the same RP for a given group because they all use a common RP hashing algorithm.

Multicast Forwarding and Reverse Path Check

With unicast routing, routers and multilayer switches forward traffic through the network along a single path from the source to the destination host whose IP address appears in the destination address field of the IP packet. Each router and switch along the way makes a unicast forwarding decision, using the destination IP address in the packet, by looking up the destination address in the unicast routing table and forwarding the packet through the specified interface to the next hop toward the destination.

With multicasting, the source is sending traffic to an arbitrary group of hosts represented by a multicast group address in the destination address field of the IP packet. To decide whether to forward or drop an incoming multicast packet, the router or multilayer switch uses a reverse path forwarding (RPF) check on the packet as follows and shown in Figure 34-1:

1. The router or multilayer switch examines the source address of the arriving multicast packet to decide whether the packet arrived on an interface that is on the reverse path back to the source.
2. If the packet arrives on the interface leading back to the source, the RPF check is successful and the packet is forwarded to all interfaces in the outgoing interface list (which might not be all interfaces on the router).
3. If the RPF check fails, the packet is discarded.

Some multicast routing protocols maintain a separate multicast routing table and use it for the RPF check. However, PIM uses the unicast routing table to perform the RPF check.

Figure 34-1 shows port 2 receiving a multicast packet from source 151.10.3.21. Table 34-1 shows that the port on the reverse path to the source is port 1, not port 2. Because the RPF check fails, the multilayer switch discards the packet. Another multicast packet from source 151.10.3.21 is received on port 1, and the routing table shows this port is on the reverse path to the source. Because the RPF check passes, the switch forwards the packet to all port in the outgoing port list.

Figure 34-1 RPF Check

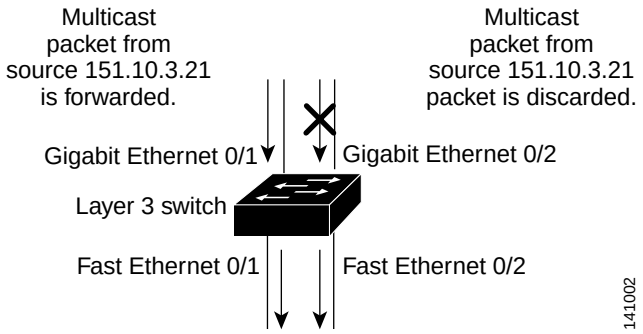


Table 34-1 Routing Table Example for an RPF Check

Network	Port
151.10.0.0/16	Gigabit Ethernet 0/1
198.14.32.0/32	Fast Ethernet 0/1
204.1.16.0/24	Fast Ethernet 0/2

PIM uses both source trees and RP-rooted shared trees to forward datagrams (described in the "PIM DM" section on page 34-4 and the "PIM SM" section on page 34-4). The RPF check is performed differently for each:

- If a PIM router or multilayer switch has a source-tree state (that is, an (S,G) entry is present in the multicast routing table), it performs the RPF check against the IP address of the source of the multicast packet.
- If a PIM router or multilayer switch has a shared-tree state (and no explicit source-tree state), it performs the RPF check on the RP address (which is known when members join the group).

Sparse-mode PIM uses the RPF lookup function to decide where it needs to send joins and prunes:

- (S,G) joins (which are source-tree states) are sent toward the source.
- (*,G) joins (which are shared-tree states) are sent toward the RP.

Dense-mode PIM uses only source trees and use RPF as previously described.

Configuring IP Multicast Routing

These sections contain this configuration information:

- Default Multicast Routing Configuration, page 34-7
- Multicast Routing Configuration Guidelines, page 34-7
- Configuring Basic Multicast Routing, page 34-9 (required)
- Configuring a Rendezvous Point, page 34-10 (required if the interface is in sparse-dense mode, and you want to treat the group as a sparse group)
- Using Auto-RP and a BSR, page 34-20 (required for non-Cisco PIMv2 devices to interoperate with Cisco PIM v1 devices))
- Monitoring the RP Mapping Information, page 34-21 (optional)
- Troubleshooting PIMv1 and PIMv2 Interoperability Problems, page 34-21 (optional)

Default Multicast Routing Configuration

Table 34-2 shows the default multicast routing configuration.

Table 34-2 Default Multicast Routing Configuration

Feature	Default Setting
Multicast routing	Disabled on all interfaces.
PIM version	Version 2.
PIM mode	No mode is defined.
PIM RP address	None configured.
PIM domain border	Disabled.
PIM multicast boundary	None.
Candidate BSRs	Disabled.
Candidate RPs	Disabled.
Shortest-path tree threshold rate	0 kbps.
PIM router query message interval	30 seconds.

Multicast Routing Configuration Guidelines

To avoid misconfiguring multicast routing on your switch, review the information in these sections:

- PIMv1 and PIMv2 Interoperability, page 34-8
- Auto-RP and BSR Configuration Guidelines, page 34-8

PIMv1 and PIMv2 Interoperability

The Cisco PIMv2 implementation provides interoperability and transition between Version 1 and Version 2, although there might be some minor problems.

You can upgrade to PIMv2 incrementally. PIM Versions 1 and 2 can be configured on different routers and multilayer switches within one network. Internally, all routers and multilayer switches on a shared media network must run the same PIM version. Therefore, if a PIMv2 device detects a PIMv1 device, the Version 2 device downgrades itself to Version 1 until all Version 1 devices have been shut down or upgraded.

PIMv2 uses the BSR to discover and announce RP-set information for each group prefix to all the routers and multilayer switches in a PIM domain. PIMv1, together with the Auto-RP feature, can perform the same tasks as the PIMv2 BSR. However, Auto-RP is a standalone protocol, separate from PIMv1, and is a proprietary Cisco protocol. PIMv2 is a standards track protocol in the IETF. We recommend that you use PIMv2. The BSR mechanism interoperates with Auto-RP on Cisco routers and multilayer switches. For more information, see the "Auto-RP and BSR Configuration Guidelines" section on page 34-8.

When PIMv2 devices interoperate with PIMv1 devices, Auto-RP should have already been deployed. A PIMv2 BSR that is also an Auto-RP mapping agent automatically advertises the RP elected by Auto-RP. That is, Auto-RP sets its single RP on every router or multilayer switch in the group. Not all routers and switches in the domain use the PIMv2 hash function to select multiple RPs.

Dense-mode groups in a mixed PIMv1 and PIMv2 region need no special configuration; they automatically interoperate.

Sparse-mode groups in a mixed PIMv1 and PIMv2 region are possible because the Auto-RP feature in PIMv1 interoperates with the PIMv2 RP feature. Although all PIMv2 devices can also use PIMv1, we recommend that the RPs be upgraded to PIMv2. To ease the transition to PIMv2, we have these recommendations:

- Use Auto-RP throughout the region.
- Configure sparse-dense mode throughout the region.

If Auto-RP is not already configured in the PIMv1 regions, configure Auto-RP. For more information, see the "Configuring Auto-RP" section on page 34-12.

Auto-RP and BSR Configuration Guidelines

There are two approaches to using PIMv2. You can use Version 2 exclusively in your network or migrate to Version 2 by employing a mixed PIM version environment.

- If your network is all Cisco routers and multilayer switches, you can use either Auto-RP or BSR.
- If you have non-Cisco routers in your network, you must use BSR.
- If you have Cisco PIMv1 and PIMv2 routers and multilayer switches and non-Cisco routers, you must use both Auto-RP and BSR. If your network includes routers from other vendors, configure the Auto-RP mapping agent and the BSR on a Cisco PIMv2 device. Ensure that no PIMv1 device is located in the path a between the BSR and a non-Cisco PIMv2 device.
- Because bootstrap messages are sent hop-by-hop, a PIMv1 device prevents these messages from reaching all routers and multilayer switches in your network. Therefore, if your network has a PIMv1 device in it and only Cisco routers and multilayer switches, it is best to use Auto-RP.
- If you have a network that includes non-Cisco routers, configure the Auto-RP mapping agent and the BSR on a Cisco PIMv2 router or multilayer switch. Ensure that no PIMv1 device is on the path between the BSR and a non-Cisco PIMv2 router.

- If you have non-Cisco PIMv2 routers that need to interoperate with Cisco PIMv1 routers and multilayer switches, both Auto-RP and a BSR are required. We recommend that a Cisco PIMv2 device be both the Auto-RP mapping agent and the BSR. For more information, see the "Using Auto-RP and a BSR" section on page 34-20.

Configuring Basic Multicast Routing

You must enable IP multicast routing and configure the PIM version and the PIM mode. Then the software can forward multicast packets, and the switch can populate its multicast routing table.

Note To enable IP multicast routing, the switch must be running the metro IP access image.

You can configure an interface to be in PIM dense mode, sparse mode, or sparse-dense mode. The switch populates its multicast routing table and forwards multicast packets it receives from its directly connected LANs according to the mode setting. You must enable PIM in one of these modes for an interface to perform IP multicast routing. Enabling PIM on an interface also enables IGMP operation on that interface.

In populating the multicast routing table, dense-mode interfaces are always added to the table. Sparse-mode interfaces are added to the table only when periodic join messages are received from downstream devices or when there is a directly connected member on the interface. When forwarding from a LAN, sparse-mode operation occurs if there is an RP known for the group. If so, the packets are encapsulated and sent toward the RP. When no RP is known, the packet is flooded in a dense-mode fashion. If the multicast traffic from a specific source is sufficient, the receiver's first-hop router might send join messages toward the source to build a source-based distribution tree.

By default, multicast routing is disabled, and there is no default mode setting. This procedure is required.

Beginning in privileged EXEC mode, follow these steps to enable IP multicasting, to configure a PIM version, and to configure a PIM mode. This procedure is required.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip multicast-routing distributed**	Enable IP multicast distributed switching.
Step 3	**interface** *interface-id*	Specify the Layer 3 interface on which you want to enable multicast routing, and enter interface configuration mode. The specified interface must be one of the following: • A routed port: a physical port that has been configured as a Layer 3 port by entering the **no switchport** interface configuration command. • An SVI: a VLAN interface created by using the **interface vlan** *vlan-id* global configuration command. These interfaces must have IP addresses assigned to them. For more information, see the "Configuring Layer 3 Interfaces" section on page 9-20.
Step 4	**no shutdown**	Enable the port, if necessary. By default, user network interfaces (UNIs) are disabled, and network node interfaces (NNIs) are enabled.

	Command	Purpose
Step 5	**ip pim version** [**1** \| **2**]	Configure the PIM version on the interface. By default, Version 2 is enabled and is the recommended setting. An interface in PIMv2 mode automatically downgrades to PIMv1 mode if that interface has a PIMv1 neighbor. The interface returns to Version 2 mode after all Version 1 neighbors are shut down or upgraded. For more information, see the "PIMv1 and PIMv2 Interoperability" section on page 34-8.
Step 6	**ip pim** {**dense-mode** \| **sparse-mode** \| **sparse-dense-mode**}	Enable a PIM mode on the interface. By default, no mode is configured. The keywords have these meanings: • **dense-mode**—Enables dense mode of operation. • **sparse-mode**—Enables sparse mode of operation. If you configure sparse-mode, you must also configure an RP. For more information, see the "Configuring a Rendezvous Point" section on page 34-10. • **sparse-dense-mode**—Causes the interface to be treated in the mode in which the group belongs. Sparse-dense-mode is the recommended setting. **Note** After you enable a PIM mode on the interface, the **ip mroute-cache distributed** interface configuration command is automatically entered for the interface and appears in the running configuration.
Step 7	**end**	Return to privileged EXEC mode.
Step 8	**show running-config**	Verify your entries.
Step 9	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable multicasting, use the **no ip multicast-routing distributed** global configuration command. To return to the default PIM version, use the **no ip pim version** interface configuration command. To disable PIM on an interface, use the **no ip pim** interface configuration command.

Configuring a Rendezvous Point

You must have an RP if the interface is in sparse-dense mode and if you want to treat the group as a sparse group. You can use several methods, as described in these sections:

- Manually Assigning an RP to Multicast Groups, page 34-11
- Configuring Auto-RP, page 34-12 (a standalone, Cisco-proprietary protocol separate from PIMv1)
- Configuring PIMv2 BSR, page 34-16 (a standards track protocol in the Internet Engineering Task Force (IETF)

You can use Auto-RP, BSR, or a combination of both, depending on the PIM version you are running and the types of routers in your network. For more information, see the "PIMv1 and PIMv2 Interoperability" section on page 34-8 and the "Auto-RP and BSR Configuration Guidelines" section on page 34-8.

Manually Assigning an RP to Multicast Groups

This section explains how to manually configure an RP. If the RP for a group is learned through a dynamic mechanism (such as Auto-RP or BSR), you need not perform this task for that RP.

Senders of multicast traffic announce their existence through register messages received from the source's first-hop router (designated router) and forwarded to the RP. Receivers of multicast packets use RPs to join a multicast group by using explicit join messages. RPs are not members of the multicast group; rather, they serve as a *meeting place* for multicast sources and group members.

You can configure a single RP for multiple groups defined by an access list. If there is no RP configured for a group, the multilayer switch treats the group as dense and uses the dense-mode PIM techniques.

Beginning in privileged EXEC mode, follow these steps to manually configure the address of the RP. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip pim rp-address** *ip-address* [*access-list-number*] [**override**]	Configure the address of a PIM RP. By default, no PIM RP address is configured. You must configure the IP address of RPs on all routers and multilayer switches (including the RP). If there is no RP configured for a group, the switch treats the group as dense, using the dense-mode PIM techniques. A PIM device can be an RP for more than one group. Only one RP address can be used at a time within a PIM domain. The access-list conditions specify for which groups the device is an RP. • For *ip-address*, enter the unicast address of the RP in dotted-decimal notation. • (Optional) For *access-list-number*, enter an IP standard access list number from 1 to 99. If no access list is configured, the RP is used for all groups. • (Optional) The **override** keyword means that if there is a conflict between the RP configured with this command and one learned by Auto-RP or BSR, the RP configured with this command prevails.
Step 3	**access-list** *access-list-number* {**deny** \| **permit**} *source* [*source-wildcard*]	Create a standard access list, repeating the command as many times as necessary. • For *access-list-number*, enter the access list number specified in Step 2. • The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. • For *source*, enter the multicast group address for which the RP should be used. • (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
Step 4	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove an RP address, use the **no ip pim rp-address** *ip-address* [*access-list-number*] [**override**] global configuration command.

This example shows how to configure the address of the RP to 147.106.6.22 for multicast group 225.2.2.2 only:

```
Switch(config)# access-list 1 permit 225.2.2.2 0.0.0.0
Switch(config)# ip pim rp-address 147.106.6.22 1
```

Configuring Auto-RP

Auto-RP uses IP multicast to automate the distribution of group-to-RP mappings to all Cisco routers and multilayer switches in a PIM network. It has these benefits:

- It is easy to use multiple RPs within a network to serve different group ranges.
- It provides load splitting among different RPs and arrangement of RPs according to the location of group participants.
- It avoids inconsistent, manual RP configurations on every router and multilayer switch in a PIM network, which can cause connectivity problems.

Note If you configure PIM in sparse mode or sparse-dense mode and do not configure Auto-RP, you must manually configure an RP as described in the "Manually Assigning an RP to Multicast Groups" section on page 34-11.

Note If routed interfaces are configured in sparse mode, Auto-RP can still be used if all devices are configured with a manual RP address for the Auto-RP groups.

These sections describe how to configure Auto-RP:

- Setting up Auto-RP in a New Internetwork, page 34-12 (optional)
- Adding Auto-RP to an Existing Sparse-Mode Cloud, page 34-13 (optional)
- Preventing Join Messages to False RPs, page 34-14 (optional)
- Filtering Incoming RP Announcement Messages, page 34-15 (optional)

For overview information, see the "Auto-RP" section on page 34-4.

Setting up Auto-RP in a New Internetwork

If you are setting up Auto-RP in a new internetwork, you do not need a default RP because you configure all the interfaces for sparse-dense mode. Follow the process described in the "Adding Auto-RP to an Existing Sparse-Mode Cloud" section on page 34-13. However, omit Step 3 if you want to configure a PIM router as the RP for the local group.

Adding Auto-RP to an Existing Sparse-Mode Cloud

This section contains some suggestions for the initial deployment of Auto-RP into an existing sparse-mode cloud to minimize disruption of the existing multicast infrastructure.

Beginning in privileged EXEC mode, follow these steps to deploy Auto-RP in an existing sparse-mode cloud. This procedure is optional.

	Command	Purpose
Step 1	**show running-config**	Verify that a default RP is already configured on all PIM devices and the RP in the sparse-mode network. It was previously configured with the **ip pim rp-address** global configuration command. This step is not required for spare-dense-mode environments. The selected RP should have good connectivity and be available across the network. Use this RP for the global groups (for example 224.x.x.x and other global groups). Do not reconfigure the group address range that this RP serves. RPs dynamically discovered through Auto-RP take precedence over statically configured RPs. Assume that it is desirable to use a second RP for the local groups.
Step 2	**configure terminal**	Enter global configuration mode.
Step 3	**ip pim send-rp-announce** *interface-id* **scope** *ttl* **group-list** *access-list-number* **interval** *seconds*	Configure another PIM device to be the candidate RP for local groups. • For *interface-id*, enter the interface type and number that identifies the RP address. Valid interfaces include physical ports, port channels, and VLANs. • For **scope** *ttl*, specify the time-to-live value in hops. Enter a hop count that is high enough so that the RP-announce messages reach all mapping agents in the network. There is no default setting. The range is 1 to 255. • For **group-list** *access-list-number*, enter an IP standard access list number from 1 to 99. If no access list is configured, the RP is used for all groups. • For **interval** *seconds*, specify how often the announcement messages must be sent. The default is 60 seconds. The range is 1 to 16383.
Step 4	**access-list** *access-list-number* {**deny** \| **permit**} *source* [*source-wildcard*]	Create a standard access list, repeating the command as many times as necessary. • For *access-list-number*, enter the access list number specified in Step 3. • The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. • For *source*, enter the multicast group address range for which the RP should be used. • (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.

	Command	Purpose
Step 5	**ip pim send-rp-discovery scope** *ttl*	Find a switch whose connectivity is not likely to be interrupted, and assign it the role of RP-mapping agent. For **scope** *ttl*, specify the time-to-live value in hops to limit the RP discovery packets. All devices within the hop count from the source device receive the Auto-RP discovery messages. These messages tell other devices which group-to-RP mapping to use to avoid conflicts (such as overlapping group-to-RP ranges). There is no default setting. The range is 1 to 255.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show running-config**	Verify your entries.
	show ip pim rp mapping	Display active RPs that are cached with associated multicast routing entries.
	show ip pim rp	Display the information cached in the routing table.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the PIM device configured as the candidate RP, use the **no ip pim send-rp-announce** *interface-id* global configuration command. To remove the switch as the RP-mapping agent, use the **no ip pim send-rp-discovery** global configuration command.

This example shows how to send RP announcements out all PIM-enabled interfaces for a maximum of 31 hops. The IP address of port 1 is the RP. Access list 5 describes the group for which this switch serves as RP:

```
Switch(config)# ip pim send-rp-announce gigabitethernet0/1 scope 31 group-list 5
Switch(config)# access-list 5 permit 224.0.0.0 15.255.255.255
```

Preventing Join Messages to False RPs

Find whether the **ip pim accept-rp** command was previously configured throughout the network by using the **show running-config** privileged EXEC command. If the **ip pim accept-rp** command is not configured on any device, this problem can be addressed later. In those routers or multilayer switches already configured with the **ip pim accept-rp** command, you must enter the command again to accept the newly advertised RP.

To accept all RPs advertised with Auto-RP and reject all other RPs by default, use the **ip pim accept-rp auto-rp** global configuration command. This procedure is optional.

If all interfaces are in sparse mode, use a default-configured RP to support the two well-known groups 224.0.1.39 and 224.0.1.40. Auto-RP uses these two well-known groups to collect and distribute RP-mapping information. When this is the case and the **ip pim accept-rp auto-rp** command is configured, another **ip pim accept-rp** command accepting the RP must be configured as follows:

```
Switch(config)# ip pim accept-rp 172.10.20.1 1
Switch(config)# access-list 1 permit 224.0.1.39
Switch(config)# access-list 1 permit 224.0.1.40
```

Filtering Incoming RP Announcement Messages

You can add configuration commands to the mapping agents to prevent a maliciously configured router from masquerading as a candidate RP and causing problems.

Beginning in privileged EXEC mode, follow these steps to filter incoming RP announcement messages. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip pim rp-announce-filter rp-list** *access-list-number* **group-list** *access-list-number*	Filter incoming RP announcement messages. Enter this command on each mapping agent in the network. Without this command, all incoming RP-announce messages are accepted by default. For **rp-list** *access-list-number*, configure an access list of candidate RP addresses that, if permitted, is accepted for the group ranges supplied in the **group-list** *access-list-number* variable. If this variable is omitted, the filter applies to all multicast groups. If more than one mapping agent is used, the filters must be consistent across all mapping agents to ensure that no conflicts occur in the Group-to-RP mapping information.
Step 3	**access-list** *access-list-number* {**deny** \| **permit**} *source* [*source-wildcard*]	Create a standard access list, repeating the command as many times as necessary. • For *access-list-number*, enter the access list number specified in Step 2. • The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. • Create an access list that specifies from which routers and multilayer switches the mapping agent accepts candidate RP announcements (rp-list ACL). • Create an access list that specifies the range of multicast groups from which to accept or deny (group-list ACL). • For *source*, enter the multicast group address range for which the RP should be used. • (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove a filter on incoming RP announcement messages, use the **no ip pim rp-announce-filter rp-list** *access-list-number* [**group-list** *access-list-number*] global configuration command.

This example shows a sample configuration on an Auto-RP mapping agent that is used to prevent candidate RP announcements from being accepted from unauthorized candidate RPs:

```
Switch(config)# ip pim rp-announce-filter rp-list 10 group-list 20
Switch(config)# access-list 10 permit host 172.16.5.1
Switch(config)# access-list 10 permit host 172.16.2.1
Switch(config)# access-list 20 deny 239.0.0.0 0.0.255.255
Switch(config)# access-list 20 permit 224.0.0.0 15.255.255.255
```

In this example, the mapping agent accepts candidate RP announcements from only two devices, 172.16.5.1 and 172.16.2.1. The mapping agent accepts candidate RP announcements from these two devices only for multicast groups that fall in the group range of 224.0.0.0 to 239.255.255.255. The mapping agent does not accept candidate RP announcements from any other devices in the network. Furthermore, the mapping agent does not accept candidate RP announcements from 172.16.5.1 or 172.16.2.1 if the announcements are for any groups in the 239.0.0.0 through 239.255.255.255 range. This range is the administratively scoped address range.

Configuring PIMv2 BSR

These sections describe how to set up BSR in your PIMv2 network:

- Defining the PIM Domain Border, page 34-16 (optional)
- Defining the IP Multicast Boundary, page 34-17 (optional)
- Configuring Candidate BSRs, page 34-18 (optional)
- Configuring Candidate RPs, page 34-19 (optional)

For overview information, see the "Bootstrap Router" section on page 34-5.

Defining the PIM Domain Border

As IP multicast becomes more widespread, the chance of one PIMv2 domain bordering another PIMv2 domain is increasing. Because these two domains probably do not share the same set of RPs, BSR, candidate RPs, and candidate BSRs, you need to constrain PIMv2 BSR messages from flowing into or out of the domain. Allowing these messages to leak across the domain borders could adversely affect the normal BSR election mechanism and elect a single BSR across all bordering domains and co-mingle candidate RP advertisements, resulting in the election of RPs in the wrong domain.

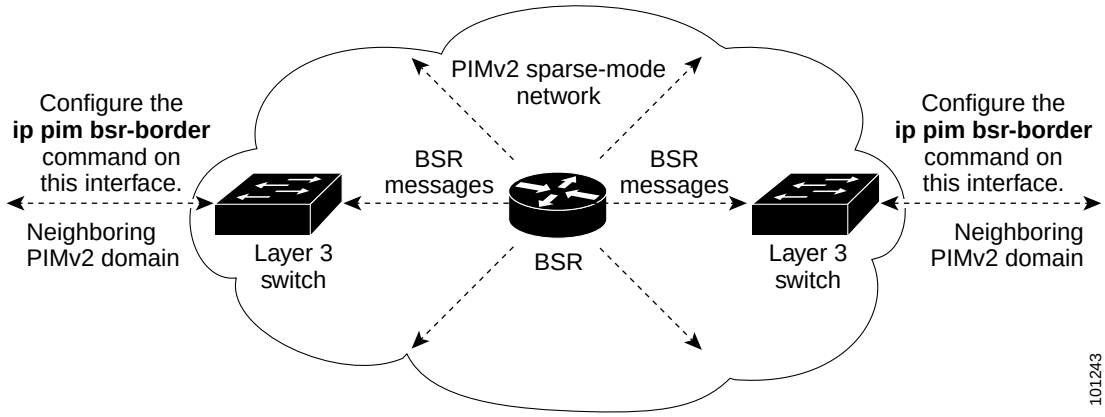
Beginning in privileged EXEC mode, follow these steps to define the PIM domain border. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled, and NNIs are enabled.
Step 4	**ip pim bsr-border**	Define a PIM bootstrap message boundary for the PIM domain. Enter this command on each interface that connects to other bordering PIM domains. This command instructs the switch to neither send or receive PIMv2 BSR messages on this interface as shown in Figure 34-2.
Step 5	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 6	**show running-config**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the PIM border, use the **no ip pim bsr-border** interface configuration command.

Figure 34-2 Constraining PIMv2 BSR Messages



Defining the IP Multicast Boundary

You define a multicast boundary to prevent Auto-RP messages from entering the PIM domain. You create an access list to deny packets destined for 224.0.1.39 and 224.0.1.40, which carry Auto-RP information.

Beginning in privileged EXEC mode, follow these steps to define a multicast boundary. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**access-list** *access-list-number* **deny** *source* [*source-wildcard*]	Create a standard access list, repeating the command as many times as necessary. • For *access-list-number*, the range is 1 to 99. • The **deny** keyword denies access if the conditions are matched. • For *source*, enter multicast addresses 224.0.1.39 and 224.0.1.40, which carry Auto-RP information. • (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
Step 3	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
Step 4	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled, and NNIs are enabled.

	Command	Purpose
Step 5	**ip multicast boundary** *access-list-number*	Configure the boundary, specifying the access list you created in Step 2.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show running-config**	Verify your entries.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the boundary, use the **no ip multicast boundary** interface configuration command.

This example shows a portion of an IP multicast boundary configuration that denies Auto-RP information:

```
Switch(config)# access-list 1 deny 224.0.1.39
Switch(config)# access-list 1 deny 224.0.1.40
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ip multicast boundary 1
```

Configuring Candidate BSRs

You can configure one or more candidate BSRs. The devices serving as candidate BSRs should have good connectivity to other devices and be in the backbone portion of the network.

Beginning in privileged EXEC mode, follow these steps to configure your switch as a candidate BSR. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip pim bsr-candidate** *interface-id hash-mask-length* [*priority*]	Configure your switch to be a candidate BSR. • For *interface-id*, enter the interface on this switch from which the BSR address is derived to make it a candidate. This interface must be enabled with PIM. Valid interfaces include physical ports, port channels, and VLANs. • For *hash-mask-length*, specify the mask length (32 bits maximum) that is to be ANDed with the group address before the hash function is called. All groups with the same seed hash correspond to the same RP. For example, if this value is 24, only the first 24 bits of the group addresses matter. • (Optional) For *priority*, enter a number from 0 to 255. The BSR with the larger priority is preferred. If the priority values are the same, the device with the highest IP address is selected as the BSR. The default is 0.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove this device as a candidate BSR, use the **no ip pim bsr-candidate** global configuration command.

This example shows how to configure a candidate BSR, which uses the IP address 172.21.24.18 on a port as the advertised BSR address, uses 30 bits as the hash-mask-length, and has a priority of 10.

```
Switch(config)# interface gigabitethernet0/2
Switch(config-if)# ip address 172.21.24.18 255.255.255.0
Switch(config-if)# ip pim sparse-dense-mode
Switch(config-if)# ip pim bsr-candidate gigabitethernet0/2 30 10
```

Configuring Candidate RPs

You can configure one or more candidate RPs. Similar to BSRs, the RPs should also have good connectivity to other devices and be in the backbone portion of the network. An RP can serve the entire IP multicast address space or a portion of it. Candidate RPs send candidate RP advertisements to the BSR. When deciding which devices should be RPs, consider these options:

- In a network of Cisco routers and multilayer switches where only Auto-RP is used, any device can be configured as an RP.
- In a network that includes only Cisco PIMv2 routers and multilayer switches and with routers from other vendors, any device can be used as an RP.
- In a network of Cisco PIMv1 routers, Cisco PIMv2 routers, and routers from other vendors, configure only Cisco PIMv2 routers and multilayer switches as RPs.

Beginning in privileged EXEC mode, follow these steps to configure your switch to advertise itself as a PIMv2 candidate RP to the BSR. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip pim rp-candidate** *interface-id* [**group-list** *access-list-number*]	Configure your switch to be a candidate RP. • For *interface-id*, specify the interface whose associated IP address is advertised as a candidate RP address. Valid interfaces include physical ports, port channels, and VLANs. • (Optional) For **group-list** *access-list-number*, enter an IP standard access list number from 1 to 99. If no group-list is specified, the switch is a candidate RP for all groups.
Step 3	**access-list** *access-list-number* {**deny** \| **permit**} *source* [*source-wildcard*]	Create a standard access list, repeating the command as many times as necessary. • For *access-list-number*, enter the access list number specified in Step 2. • The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. • For *source*, enter the number of the network or host from which the packet is being sent. • (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
Step 4	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove this device as a candidate RP, use the **no ip pim rp-candidate** *interface-id* global configuration command.

This example shows how to configure the switch to advertise itself as a candidate RP to the BSR in its PIM domain. Standard access list number 4 specifies the group prefix associated with the RP that has the address identified by a port. That RP is responsible for the groups with the prefix 239.

```
Switch(config)# ip pim rp-candidate gigabitethernet0/2 group-list 4
Switch(config)# access-list 4 permit 239.0.0.0 0.255.255.255
```

Using Auto-RP and a BSR

If there are only Cisco devices in you network (no routers from other vendors), there is no need to configure a BSR. Configure Auto-RP in a network that is running both PIMv1 and PIMv2.

If you have non-Cisco PIMv2 routers that need to interoperate with Cisco PIMv1 routers and multilayer switches, both Auto-RP and a BSR are required. We recommend that a Cisco PIMv2 router or multilayer switch be both the Auto-RP mapping agent and the BSR.

If you must have one or more BSRs, we have these recommendations:

- Configure the candidate BSRs as the RP-mapping agents for Auto-RP. For more information, see the "Configuring Auto-RP" section on page 34-12 and the "Configuring Candidate BSRs" section on page 34-18.
- For group prefixes advertised through Auto-RP, the PIMv2 BSR mechanism should not advertise a subrange of these group prefixes served by a different set of RPs. In a mixed PIMv1 and PIMv2 domain, have backup RPs serve the same group prefixes. This prevents the PIMv2 DRs from selecting a different RP from those PIMv1 DRs, due to the longest match lookup in the RP-mapping database.

Beginning in privileged EXEC mode, follow these steps to verify the consistency of group-to-RP mappings. This procedure is optional.

	Command	Purpose
Step 1	**show ip pim rp** [[*group-name* \| *group-address*] \| **mapping**]	On any Cisco device, display the available RP mappings. • (Optional) For *group-name*, specify the name of the group about which to display RPs. • (Optional) For *group-address*, specify the address of the group about which to display RPs. • (Optional) Use the **mapping** keyword to display all group-to-RP mappings of which the Cisco device is aware (either configured or learned from Auto-RP).
Step 2	**show ip pim rp-hash** *group*	On a PIMv2 router or multilayer switch, confirm that the same RP is the one that a PIMv1 system chooses. For *group*, enter the group address for which to display RP information.

Monitoring the RP Mapping Information

To monitor the RP mapping information, use these commands in privileged EXEC mode:

- **show ip pim bsr** displays information about the elected BSR.
- **show ip pim rp-hash** *group* displays the RP that was selected for the specified group.
- **show ip pim rp** [*group-name* | *group-address* | **mapping**] displays how the switch learns of the RP (through the BSR or the Auto-RP mechanism).

Troubleshooting PIMv1 and PIMv2 Interoperability Problems

When debugging interoperability problems between PIMv1 and PIMv2, check these in the order shown:

1. Verify RP mapping with the **show ip pim rp-hash** privileged EXEC command, making sure that all systems agree on the same RP for the same group.
2. Verify interoperability between different versions of DRs and RPs. Make sure the RPs are interacting with the DRs properly (by responding with register-stops and forwarding decapsulated data packets from registers).

Configuring Advanced PIM Features

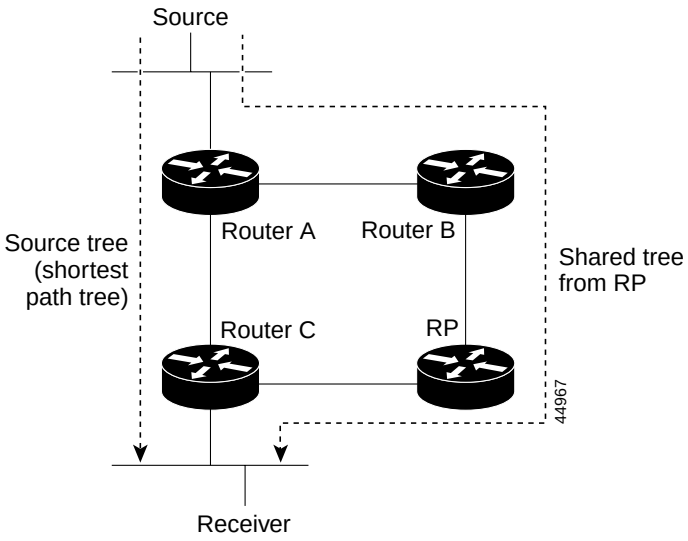
These sections describe the optional advanced PIM features:

- Understanding PIM Shared Tree and Source Tree, page 34-21
- Delaying the Use of PIM Shortest-Path Tree, page 34-23 (optional)
- Modifying the PIM Router-Query Message Interval, page 34-24 (optional)

Understanding PIM Shared Tree and Source Tree

By default, members of a group receive data from senders to the group across a single data-distribution tree rooted at the RP. Figure 34-3 shows this type of shared-distribution tree. Data from senders is delivered to the RP for distribution to group members joined to the shared tree.

Figure 34-3 Shared Tree and Source Tree (Shortest-Path Tree)



If the data rate warrants, leaf routers (routers without any downstream connections) on the shared tree can use the data distribution tree rooted at the source. This type of distribution tree is called a shortest-path tree or source tree. By default, the software switches to a source tree upon receiving the first data packet from a source.

This process describes the move from a shared tree to a source tree:

1. A receiver joins a group; leaf Router C sends a join message toward the RP.
2. The RP puts a link to Router C in its outgoing interface list.
3. A source sends data; Router A encapsulates the data in a register message and sends it to the RP.
4. The RP forwards the data down the shared tree to Router C and sends a join message toward the source. At this point, data might arrive twice at Router C, once encapsulated and once natively.
5. When data arrives natively (unencapsulated) at the RP, it sends a register-stop message to Router A.
6. By default, reception of the first data packet prompts Router C to send a join message toward the source.
7. When Router C receives data on (S,G), it sends a prune message for the source up the shared tree.
8. The RP deletes the link to Router C from the outgoing interface of (S,G). The RP triggers a prune message toward the source.

Join and prune messages are sent for sources and RPs. They are sent hop-by-hop and are processed by each PIM device along the path to the source or RP. Register and register-stop messages are not sent hop-by-hop. They are sent by the designated router that is directly connected to a source and are received by the RP for the group.

Multiple sources sending to groups use the shared tree.

You can configure the PIM device to stay on the shared tree. For more information, see the "Delaying the Use of PIM Shortest-Path Tree" section on page 34-23.

Delaying the Use of PIM Shortest-Path Tree

The change from shared to source tree happens when the first data packet arrives at the last-hop router (Router C in Figure 34-3). This change occurs because the **ip pim spt-threshold** global configuration command controls that timing.

The shortest-path tree requires more memory than the shared tree but reduces delay. You might want to postpone its use. Instead of allowing the leaf router to immediately move to the shortest-path tree, you can specify that the traffic must first reach a threshold.

You can configure when a PIM leaf router should join the shortest-path tree for a specified group. If a source sends at a rate greater than or equal to the specified kbps rate, the multilayer switch triggers a PIM join message toward the source to construct a source tree (shortest-path tree). If the traffic rate from the source drops below the threshold value, the leaf router switches back to the shared tree and sends a prune message toward the source.

You can specify to which groups the shortest-path tree threshold applies by using a group list (a standard access list). If a value of 0 is specified or if the group list is not used, the threshold applies to all groups.

Beginning in privileged EXEC mode, follow these steps to configure a traffic rate threshold that must be reached before multicast routing is switched from the source tree to the shortest-path tree. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**access-list** *access-list-number* {**deny** \| **permit**} *source* [*source-wildcard*]	Create a standard access list. • For *access-list-number*, the range is 1 to 99. • The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. • For *source*, specify the multicast group to which the threshold will apply. • (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
Step 3	**ip pim spt-threshold** {*kbps* \| **infinity**} [**group-list** *access-list-number*]	Specify the threshold that must be reached before moving to shortest-path tree (spt). • For *kbps*, specify the traffic rate in kilobits per second. The default is 0 kbps. **Note** Because of switch hardware limitations, 0 kbps is the only valid entry even though the range is 0 to 4294967. • Specify **infinity** if you want all sources for the specified group to use the shared tree, never switching to the source tree. • (Optional) For **group-list** *access-list-number*, specify the access list created in Step 2. If the value is 0 or if the group-list is not used, the threshold applies to all groups.

	Command	Purpose
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no ip pim spt-threshold** {*kbps* | **infinity**} global configuration command.

Modifying the PIM Router-Query Message Interval

PIM routers and multilayer switches send PIM router-query messages to find which device will be the DR for each LAN segment (subnet). The DR is responsible for sending IGMP host-query messages to all hosts on the directly connected LAN.

With PIM DM operation, the DR has meaning only if IGMPv1 is in use. IGMPv1 does not have an IGMP querier election process, so the elected DR functions as the IGMP querier. With PIM SM operation, the DR is the device that is directly connected to the multicast source. It sends PIM register messages to notify the RP that multicast traffic from a source needs to be forwarded down the shared tree. In this case, the DR is the device with the highest IP address.

Beginning in privileged EXEC mode, follow these steps to modify the router-query message interval. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled, and NNIs are enabled.
Step 4	**ip pim query-interval** *seconds*	Configure the frequency at which the switch sends PIM router-query messages. The default is 30 seconds. The range is 1 to 65535.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show ip igmp interface** [*interface-id*]	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no ip pim query-interval** [*seconds*] interface configuration command.

Configuring Optional IGMP Features

These sections contain this configuration information:

- Default IGMP Configuration, page 34-25
- Configuring the Switch as a Member of a Group, page 34-25 (optional)
- Controlling Access to IP Multicast Groups, page 34-26 (optional)
- Changing the IGMP Version, page 34-27 (optional)
- Modifying the IGMP Host-Query Message Interval, page 34-28 (optional)
- Changing the IGMP Query Timeout for IGMPv2, page 34-29 (optional)
- Changing the Maximum Query Response Time for IGMPv2, page 34-29 (optional)
- Configuring the Switch as a Statically Connected Member, page 34-30 (optional)

Default IGMP Configuration

Table 34-3 shows the default IGMP configuration.

Table 34-3 Default IGMP Configuration

Feature	Default Setting
Multilayer switch as a member of a multicast group	No group memberships are defined.
Access to multicast groups	All groups are allowed on an interface.
IGMP version	Version 2 on all interfaces.
IGMP host-query message interval	60 seconds on all interfaces.
IGMP query timeout	60 seconds on all interfaces.
IGMP maximum query response time	10 seconds on all interfaces.
Multilayer switch as a statically connected member	Disabled.

Configuring the Switch as a Member of a Group

You can configure the switch as a member of a multicast group and discover multicast reachability in a network. If all the multicast-capable routers and multilayer switches that you administer are members of a multicast group, pinging that group causes all these devices to respond. The devices respond to IGMP echo-request packets addressed to a group of which they are members. Another example is the multicast trace-route tools provided in the software.

Caution Performing this procedure might impact the CPU performance because the CPU will receive all data traffic for the group address.

Beginning in privileged EXEC mode, follow these steps to configure the switch to be a member of a group. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled, and NNIs are enabled.
Step 4	**ip igmp join-group** *group-address*	Configure the switch to join a multicast group. By default, no group memberships are defined. For *group-address*, specify the multicast IP address in dotted decimal notation.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show ip igmp interface** [*interface-id*]	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To cancel membership in a group, use the **no ip igmp join-group** *group-address* interface configuration command.

This example shows how to enable the switch to join multicast group 255.2.2.2:

```
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ip igmp join-group 255.2.2.2
```

Controlling Access to IP Multicast Groups

The switch sends IGMP host-query messages to find which multicast groups have members on attached local networks. The switch then forwards to these group members all packets addressed to the multicast group. You can place a filter on each interface to restrict the multicast groups that hosts on the subnet serviced by the interface can join.

Beginning in privileged EXEC mode, follow these steps to filter multicast groups allowed on an interface. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled, and NNIs are enabled.
Step 4	**ip igmp access-group** *access-list-number*	Specify the multicast groups that hosts on the subnet serviced by an interface can join. By default, all groups are allowed on an interface. For *access-list-number*, specify an IP standard access list number. The range is 1 to 99.

	Command	Purpose
Step 5	**exit**	Return to global configuration mode.
Step 6	**access-list** *access-list-number* {**deny** \| **permit**} *source* [*source-wildcard*]	Create a standard access list. • For *access-list-number*, specify the access list created in Step 3. • The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. • For *source*, specify the multicast group that hosts on the subnet can join. • (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
Step 7	**end**	Return to privileged EXEC mode.
Step 8	**show ip igmp interface** [*interface-id*]	Verify your entries.
Step 9	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable groups on an interface, use the **no ip igmp access-group** interface configuration command.

This example shows how to configure hosts attached to a port as able to join only group 255.2.2.2:

```
Switch(config)# access-list 1 255.2.2.2 0.0.0.0
Switch(config-if)# interface gigabitethernet0/1
Switch(config-if)# ip igmp access-group 1
```

Changing the IGMP Version

By default, the switch uses IGMP Version 2, which provides features such as the IGMP query timeout and the maximum query response time.

All systems on the subnet must support the same version. The switch does not automatically detect Version 1 systems and switch to Version 1. You can mix Version 1 and Version 2 hosts on the subnet because Version 2 routers or switches always work correctly with IGMPv1 hosts.

Configure the switch for Version 1 if your hosts do not support Version 2.

Beginning in privileged EXEC mode, follow these steps to change the IGMP version. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled, and NNIs are enabled.

	Command	Purpose
Step 4	**ip igmp version** {**1** \| **2**}	Specify the IGMP version that the switch uses. **Note** If you change to Version 1, you cannot configure the **ip igmp query-interval** or the **ip igmp query-max-response-time** interface configuration commands.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show ip igmp interface** [*interface-id*]	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no ip igmp version** interface configuration command.

Modifying the IGMP Host-Query Message Interval

The switch periodically sends IGMP host-query messages to discover which multicast groups are present on attached networks. These messages are sent to the all-hosts multicast group (224.0.0.1) with a time-to-live (TTL) of 1. The switch sends host-query messages to refresh its knowledge of memberships present on the network. If, after some number of queries, the software discovers that no local hosts are members of a multicast group, the software stops forwarding multicast packets to the local network from remote origins for that group and sends a prune message upstream toward the source.

The switch elects a PIM designated router (DR) for the LAN (subnet). The DR is the router or multilayer switch with the highest IP address for IGMPv2. For IGMPv1, the DR is elected according to the multicast routing protocol that runs on the LAN. The designated router is responsible for sending IGMP host-query messages to all hosts on the LAN. In sparse mode, the designated router also sends PIM register and PIM join messages toward the RP router.

Beginning in privileged EXEC mode, follow these steps to modify the host-query interval. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled, and NNIs are enabled.
Step 4	**ip igmp query-interval** *seconds*	Configure the frequency at which the designated router sends IGMP host-query messages. By default, the designated router sends IGMP host-query messages every 60 seconds to keep the IGMP overhead very low on hosts and networks. The range is 1 to 65535.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show ip igmp interface** [*interface-id*]	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no ip igmp query-interval** interface configuration command.

Changing the IGMP Query Timeout for IGMPv2

If you are using IGMPv2, you can specify the period of time before the switch takes over as the querier for the interface. By default, the switch waits twice the query interval controlled by the **ip igmp query-interval** interface configuration command. After that time, if the switch has received no queries, it becomes the querier.

You can configure the query interval by entering the **show ip igmp interface** *interface-id* privileged EXEC command.

Beginning in privileged EXEC mode, follow these steps to change the IGMP query timeout. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled, and NNIs are enabled.
Step 4	**ip igmp querier-timeout** *seconds*	Specify the IGMP query timeout. The default is 60 seconds (twice the query interval). The range is 60 to 300.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show ip igmp interface** [*interface-id*]	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no ip igmp querier-timeout** interface configuration command.

Changing the Maximum Query Response Time for IGMPv2

If you are using IGMPv2, you can change the maximum query response time advertised in IGMP queries. The maximum query response time enables the switch to quickly detect that there are no more directly connected group members on a LAN. Decreasing the value enables the switch to prune groups faster.

Beginning in privileged EXEC mode, follow these steps to change the maximum query response time. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled, and NNIs are enabled.
Step 4	**ip igmp query-max-response-time** *seconds*	Change the maximum query response time advertised in IGMP queries. The default is 10 seconds. The range is 1 to 25.

	Command	Purpose
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show ip igmp interface** [*interface-id*]	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no ip igmp query-max-response-time** interface configuration command.

Configuring the Switch as a Statically Connected Member

Sometimes there is either no group member on a network segment or a host cannot report its group membership by using IGMP. However, you might want multicast traffic to go to that network segment. These are ways to pull multicast traffic down to a network segment:

- Use the **ip igmp join-group** interface configuration command. With this method, the switch accepts the multicast packets in addition to forwarding them. Accepting the multicast packets prevents the switch from fast switching.
- Use the **ip igmp static-group** interface configuration command. With this method, the switch does not accept the packets itself, but only forwards them. This method enables fast switching. The outgoing interface appears in the IGMP cache, but the switch itself is not a member, as evidenced by lack of an *L* (local) flag in the multicast route entry.

Beginning in privileged EXEC mode, follow these steps to configure the switch itself to be a statically connected member of a group (and enable fast switching). This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled, and NNIs are enabled.
Step 4	**ip igmp static-group** *group-address*	Configure the switch as a statically connected member of a group. By default, this feature is disabled.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show ip igmp interface** [*interface-id*]	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the switch as a member of the group, use the **no ip igmp static-group** *group-address* interface configuration command.

Configuring Optional Multicast Routing Features

These sections describe how to configure optional multicast routing features:

- Configuring sdr Listener Support, page 34-31 (optional)—for MBONE multimedia conference session and set up
- Configuring an IP Multicast Boundary, page 34-32 (optional)—to control bandwidth utilization.

Configuring sdr Listener Support

The MBONE is the small subset of Internet routers and hosts that are interconnected and capable of forwarding IP multicast traffic. Other multimedia content is often broadcast over the MBONE. Before you can join a multimedia session, you need to know what multicast group address and port are being used for the session, when the session is going to be active, and what sort of applications (audio, video, and so forth) are required on your workstation. The MBONE Session Directory Version 2 (sdr) tool provides this information. This freeware application can be downloaded from several sites on the World Wide Web, one of which is http://www.video.ja.net/mice/index.html.

SDR is a multicast application that listens to a well-known multicast group address and port for Session Announcement Protocol (SAP) multicast packets from SAP clients, which announce their conference sessions. These SAP packets contain a session description, the time the session is active, its IP multicast group addresses, media format, contact person, and other information about the advertised multimedia session. The information in the SAP packet is displayed in the SDR Session Announcement window.

Enabling sdr Listener Support

By default, the switch does not listen to session directory advertisements.

Beginning in privileged EXEC mode, follow these steps to enable the switch to join the default session directory group (224.2.127.254) on the interface and listen to session directory advertisements. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to be enabled for sdr, and enter interface configuration mode.
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled, and NNIs are enabled.
Step 4	**ip sdr listen**	Enable sdr listener support.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable sdr support, use the **no ip sdr listen** interface configuration command.

Limiting How Long an sdr Cache Entry Exists

By default, entries are never deleted from the sdr cache. You can limit how long the entry remains active so that if a source stops advertising SAP information, old advertisements are not needlessly kept.

Beginning in privileged EXEC mode, follow these steps to limit how long an sdr cache entry stays active in the cache. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip sdr cache-timeout** *minutes*	Limit how long an sdr cache entry stays active in the cache. By default, entries are never deleted from the cache. For *minutes*, the range is 1 to 4294967295.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no ip sdr cache-timeout** global configuration command. To delete the entire cache, use the **clear ip sdr** privileged EXEC command.

To display the session directory cache, use the **show ip sdr** privileged EXEC command.

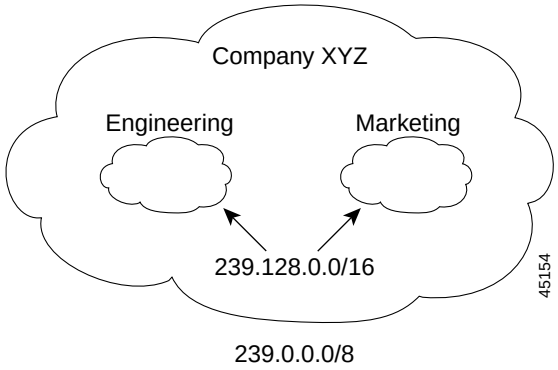
Configuring an IP Multicast Boundary

Administratively-scoped boundaries can be used to limit the forwarding of multicast traffic outside of a domain or subdomain. This approach uses a special range of multicast addresses, called *administratively-scoped addresses*, as the boundary mechanism. If you configure an administratively-scoped boundary on a routed interface, multicast traffic whose multicast group addresses fall in this range can not enter or exit this interface, thereby providing a firewall for multicast traffic in this address range.

Note Multicast boundaries and TTL thresholds control the scoping of multicast domains; however, TTL thresholds are not supported by the switch. You should use multicast boundaries instead of TTL thresholds to limit the forwarding of multicast traffic outside of a domain or a subdomain.

Figure 34-4 shows that Company XYZ has an administratively-scoped boundary set for the multicast address range 239.0.0.0/8 on all routed interfaces at the perimeter of its network. This boundary prevents any multicast traffic in the range 239.0.0.0 through 239.255.255.255 from entering or leaving the network. Similarly, the engineering and marketing departments have an administratively-scoped boundary of 239.128.0.0/16 around the perimeter of their networks. This boundary prevents multicast traffic in the range of 239.128.0.0 through 239.128.255.255 from entering or leaving their respective networks.

Figure 34-4 Administratively-Scoped Boundaries



You can define an administratively-scoped boundary on a routed interface for multicast group addresses. A standard access list defines the range of addresses affected. When a boundary is defined, no multicast data packets are allowed to flow across the boundary from either direction. The boundary allows the same multicast group address to be reused in different administrative domains.

The IANA has designated the multicast address range 239.0.0.0 to 239.255.255.255 as the administratively-scoped addresses. This range of addresses can then be reused in domains administered by different organizations. The addresses would be considered local, not globally unique.

Beginning in privileged EXEC mode, follow these steps to set up an administratively-scoped boundary. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**access-list** *access-list-number* {**deny** \| **permit**} *source* [*source-wildcard*]	Create a standard access list, repeating the command as many times as necessary. • For *access-list-number*, the range is 1 to 99. • The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. • For *source*, enter the number of the network or host from which the packet is being sent. • (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
Step 3	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
Step 4	**no shutdown**	Enable the port, if necessary. By default, UNIs are disabled, and NNIs are enabled.
Step 5	**ip multicast boundary** *access-list-number*	Configure the boundary, specifying the access list you created in Step 2.
Step 6	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 7	**show running-config**	Verify your entries.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the boundary, use the **no ip multicast boundary** interface configuration command.

This example shows how to set up a boundary for all administratively-scoped addresses:

```
Switch(config)# access-list 1 deny 239.0.0.0 0.255.255.255
Switch(config)# access-list 1 permit 224.0.0.0 15.255.255.255
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ip multicast boundary 1
```

Monitoring and Maintaining IP Multicast Routing

These sections describe how to monitor and maintain IP multicast routing:

- Clearing Caches, Tables, and Databases, page 34-34
- Displaying System and Network Statistics, page 34-34
- Monitoring IP Multicast Routing, page 34-35

Clearing Caches, Tables, and Databases

You can remove all contents of a particular cache, table, or database. Clearing a cache, table, or database might be necessary when the contents of the particular structure are or suspected to be invalid.

You can use any of the privileged EXEC commands in Table 34-4 to clear IP multicast caches, tables, and databases:

Table 34-4 Commands for Clearing Caches, Tables, and Databases

Command	Purpose
clear ip igmp group [*group-name* \| *group-address* \| *interface*]	Delete entries from the IGMP cache.
clear ip mroute {***** \| *group* [*source*]}	Delete entries from the IP multicast routing table.
clear ip pim auto-rp *rp-address*	Clear the Auto-RP cache.
clear ip sdr [*group-address* \| **"***session-name***"**]	Delete the Session Directory Protocol Version 2 cache or an sdr cache entry.

Displaying System and Network Statistics

You can display specific statistics, such as the contents of IP routing tables, caches, and databases.

Note This release does not support per-route statistics.

You can display information to learn resource utilization and solve network problems. You can also display information about node reachability and discover the routing path your device's packets are taking through the network.

You can use any of the privileged EXEC commands in Table 34-5 to display various routing statistics:

Table 34-5 Commands for Displaying System and Network Statistics

Command	Purpose
ping [*group-name* \| *group-address*]	Send an ICMP Echo Request to a multicast group address.
show ip igmp groups [*group-name* \| *group-address* \| *type number*]	Display the multicast groups that are directly connected to the switch and that were learned through IGMP.
show ip igmp interface [*type number*]	Display multicast-related information about an interface.
show ip mcache [*group* [*source*]]	Display the contents of the IP fast-switching cache.
show ip mpacket [*source-address* \| *name*] [*group-address* \| *name*] [**detail**]	Display the contents of the circular cache-header buffer.
show ip mroute [*group-name* \| *group-address*] [*source*] [**summary**] [**count**] [**active** *kbps*]	Display the contents of the IP multicast routing table.
show ip pim interface [*type number*] [**count**]	Display information about interfaces configured for PIM.
show ip pim neighbor [*type number*]	List the PIM neighbors discovered by the switch.
show ip pim rp [*group-name* \| *group-address*]	Display the RP routers associated with a sparse-mode multicast group.
show ip rpf {*source-address* \| *name*}	Display how the switch is doing Reverse-Path Forwarding (that is, from the unicast routing table or static mroutes).
show ip sdr [*group* \| **"***session-name***"** \| **detail**]	Display the Session Directory Protocol Version 2 cache.

Monitoring IP Multicast Routing

You can use the privileged EXEC commands in Table 34-6 to monitor IP multicast routers, packets, and paths:

Table 34-6 Commands for Monitoring IP Multicast Routing

Command	Purpose
mrinfo [*hostname* \| *address*] [*source-address* \| *interface*]	Query a multicast router or multilayer switch about which neighboring multicast devices are peering with it.
mstat *source* [*destination*] [*group*]	Display IP multicast packet rate and loss information.
mtrace *source* [*destination*] [*group*]	Trace the path from a source to a destination branch for a multicast distribution tree for a given group.



CHAPTER 35

Configuring MSDP

This chapter describes how to configure the Multicast Source Discovery Protocol (MSDP) on the Cisco ME 3400 Ethernet Access switch. The MSDP connects multiple Protocol-Independent Multicast sparse-mode (PIM-SM) domains.

MSDP is not fully supported in this software release because of a lack of support for Multicast Border Gateway Protocol (MBGP), which works closely with MSDP. However, it is possible to create default peers that MSDP can operate with if MBGP is not running.

To use this feature, the switch must be running the metro IP access image.

Note For complete syntax and usage information for the commands used in this chapter, see the *Cisco IOS IP Command Reference, Volume 3 of 3: Multicast, Release 12.2*.

This chapter consists of these sections:

- Understanding MSDP, page 35-1
- Configuring MSDP, page 35-4
- Monitoring and Maintaining MSDP, page 35-19

Understanding MSDP

MSDP allows multicast sources for a group to be known to all rendezvous points (RPs) in different domains. Each PIM-SM domain uses its own RPs and does not depend on RPs in other domains. An RP runs MSDP over the Transmission Control Protocol (TCP) to discover multicast sources in other domains.

An RP in a PIM-SM domain has an MSDP peering relationship with MSDP-enabled devices in another domain. The peering relationship occurs over a TCP connection, primarily exchanging a list of sources sending to multicast groups. The TCP connections between RPs are achieved by the underlying routing system. The receiving RP uses the source lists to establish a source path.

The purpose of this topology is to have domains discover multicast sources in other domains. If the multicast sources are of interest to a domain that has receivers, multicast data is delivered over the normal, source-tree building mechanism in PIM-SM. MSDP is also used to announce sources sending to a group. These announcements must originate at the domain's RP.

MSDP depends heavily on the Border Gateway Protocol (BGP) or MBGP for interdomain operation. We recommend that you run MSDP in RPs in your domain that are RPs for sources sending to global groups to be announced to the Internet.

MSDP Operation

Figure 35-1 shows MSDP operating between two MSDP peers. PIM uses MSDP as the standard mechanism to register a source with the RP of a domain. When MSDP is configured, this sequence occurs.

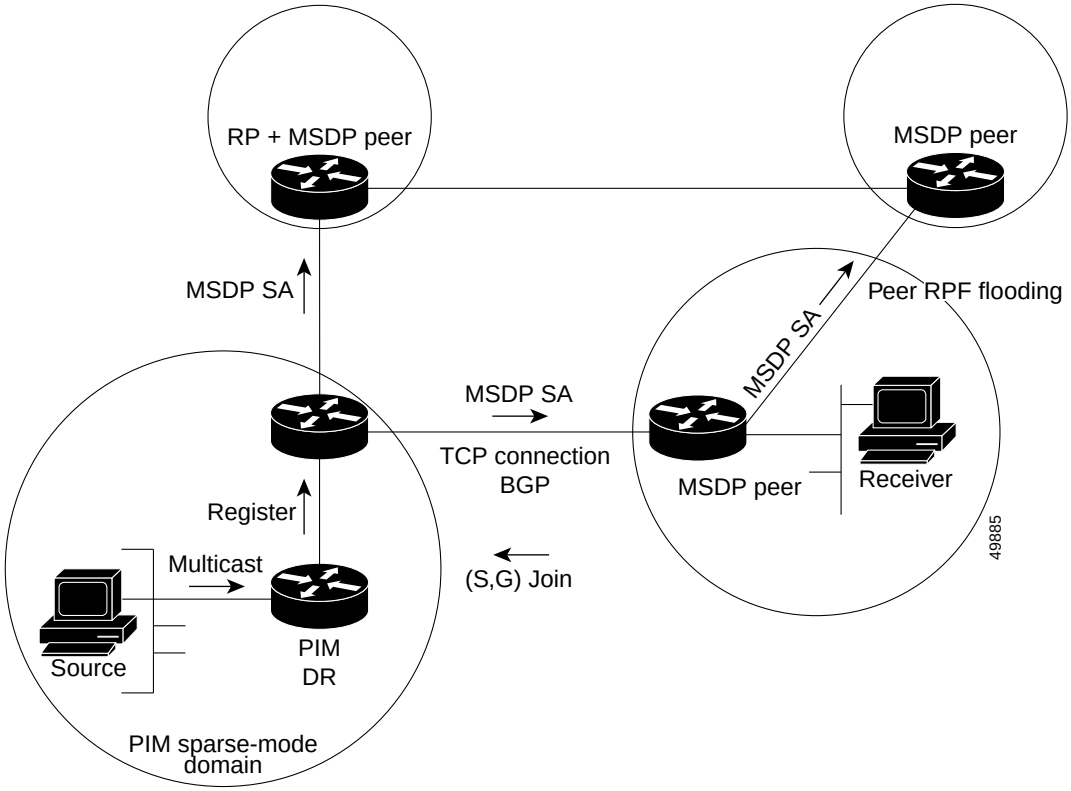
When a source sends its first multicast packet, the first-hop router (*designated router* or RP) directly connected to the source sends a PIM register message to the RP. The RP uses the register message to register the active source and to forward the multicast packet down the shared tree in the local domain. With MSDP configured, the RP also forwards a source-active (SA) message to all MSDP peers. The SA message identifies the source, the group the source is sending to, and the address of the RP or the originator ID (the IP address of the interface used as the RP address), if configured.

Each MSDP peer receives and forwards the SA message away from the originating RP to achieve peer reverse-path flooding (RPF). The MSDP device examines the BGP or MBGP routing table to discover which peer is the next hop toward the originating RP of the SA message. Such a peer is called an *RPF peer* (reverse-path forwarding peer). The MSDP device forwards the message to all MSDP peers other than the RPF peer. For information on how to configure an MSDP peer when BGP and MBGP are not supported, see the "Configuring a Default MSDP Peer" section on page 35-4.

If the MSDP peer receives the same SA message from a non-RPF peer toward the originating RP, it drops the message. Otherwise, it forwards the message to all its MSDP peers.

The RP for a domain receives the SA message from an MSDP peer. If the RP has any join requests for the group the SA message describes and if the (*,G) entry exists with a nonempty outgoing interface list, the domain is interested in the group, and the RP triggers an (S,G) join toward the source. After the (S,G) join reaches the source's DR, a branch of the source tree has been built from the source to the RP in the remote domain. Multicast traffic can now flow from the source across the source tree to the RP and then down the shared tree in the remote domain to the receiver.

Figure 35-1 MSDP Running Between RP Peers



MSDP Benefits

MSDP has these benefits:

- It breaks up the shared multicast distribution tree. You can make the shared tree local to your domain. Your local members join the local tree, and join messages for the shared tree never need to leave your domain.
- PIM sparse-mode domains can rely only on their own RPs, decreasing reliance on RPs in another domain. This increases security because you can prevent your sources from being known outside your domain.
- Domains with only receivers can receive data without globally advertising group membership.
- Global source multicast routing table state is not required, saving memory.

Configuring MSDP

These sections contain this configuration information:

- Default MSDP Configuration, page 35-4
- Configuring a Default MSDP Peer, page 35-4 (required)
- Caching Source-Active State, page 35-6 (optional)
- Requesting Source Information from an MSDP Peer, page 35-8 (optional)
- Controlling Source Information that Your Switch Originates, page 35-8 (optional)
- Controlling Source Information that Your Switch Forwards, page 35-12 (optional)
- Controlling Source Information that Your Switch Receives, page 35-14 (optional)
- Configuring an MSDP Mesh Group, page 35-16 (optional)
- Shutting Down an MSDP Peer, page 35-16 (optional)
- Including a Bordering PIM Dense-Mode Region in MSDP, page 35-17 (optional)
- Configuring an Originating Address other than the RP Address, page 35-18 (optional)

Default MSDP Configuration

MSDP is not enabled, and no default MSDP peer exists.

Configuring a Default MSDP Peer

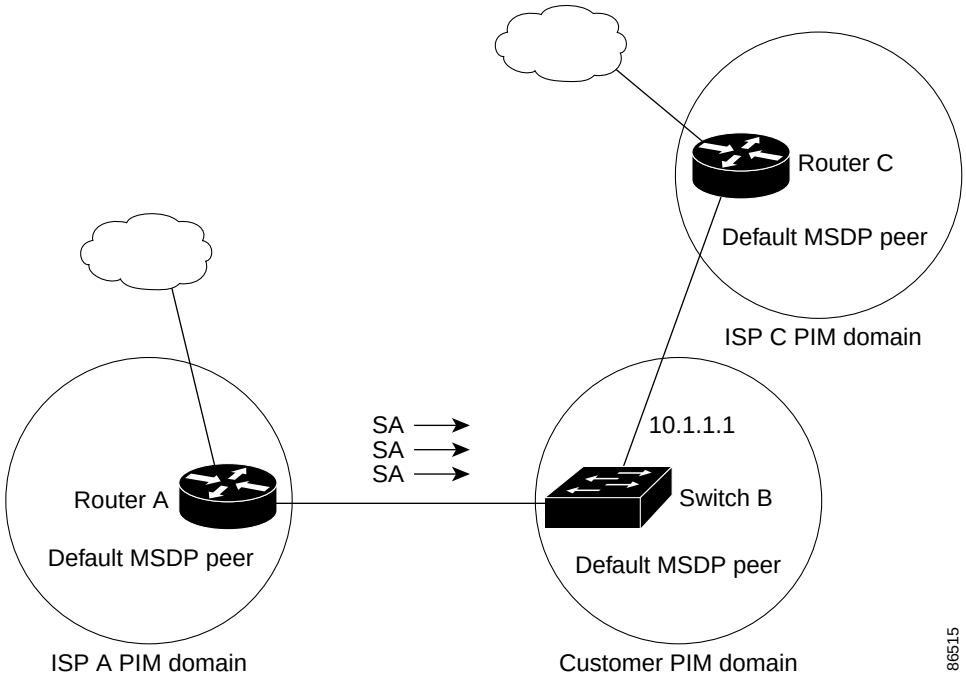
In this software release, because BGP and MBGP are not supported, you cannot configure an MSDP peer on the local switch by using the **ip msdp peer** global configuration command. Instead, you define a default MSDP peer (by using the **ip msdp default-peer** global configuration command) from which to accept all SA messages for the switch. The default MSDP peer must be a previously configured MSDP peer. Configure a default MSDP peer when the switch is not BGP- or MBGP-peering with an MSDP peer. If a single MSDP peer is configured, the switch always accepts all SA messages from that peer.

Figure 35-2 shows a network in which default MSDP peers might be used. In Figure 35-2, a customer who owns Switch B is connected to the Internet through two Internet service providers (ISPs), one owning Router A and the other owning Router C. They are not running BGP or MBGP between them. To learn about sources in the ISP's domain or in other domains, Switch B at the customer site identifies Router A as its default MSDP peer. Switch B advertises SA messages to both Router A and Router C but accepts SA messages only from Router A or only from Router C. If Router A is first in the configuration file, it is used if it is running. If Router A is not running, only then does Switch B accept SA messages from Router C. This is the default behavior without a prefix list.

If you specify a prefix list, the peer is a default peer only for the prefixes in the list. You can have multiple active default peers when you have a prefix list associated with each. When you do not have any prefix lists, you can configure multiple default peers, but only the first one is the active default peer as long as the router has connectivity to this peer and the peer is alive. If the first configured peer fails or the connectivity to this peer fails, the second configured peer becomes the active default, and so on.

The ISP probably uses a prefix list to define which prefixes it accepts from the customer's router.

Figure 35-2 Default MSDP Peer Network



Beginning in privileged EXEC mode, follow these steps to specify a default MSDP peer. This procedure is required.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip msdp default-peer** *ip-address* \| *name* [**prefix-list** *list*]	Define a default peer from which to accept all MSDP SA messages. • For *ip-address* \| *name*, enter the IP address or Domain Name System (DNS) server name of the MSDP default peer. • (Optional) For **prefix-list** *list*, enter the list name that specifies the peer to be the default peer only for the listed prefixes. You can have multiple active default peers when you have a prefix list associated with each. When you enter multiple **ip msdp default-peer** commands with the **prefix-list** keyword, you use all the default peers at the same time for different RP prefixes. This syntax is typically used in a service provider cloud that connects stub site clouds. When you enter multiple **ip msdp default-peer** commands without the **prefix-list** keyword, a single active peer accepts all SA messages. If that peer fails, the next configured default peer accepts all SA messages. This syntax is typically used at a stub site.

	Command	Purpose
Step 3	**ip prefix-list** *name* [**description** *string*] \| **seq** *number* {**permit** \| **deny**} *network length*	(Optional) Create a prefix list using the name specified in Step 2. • (Optional) For **description** *string*, enter a description of up to 80 characters to describe this prefix list. • For **seq** *number*, enter the sequence number of the entry. The range is 1 to 4294967294. • The **deny** keyword denies access to matching conditions. • The **permit** keyword permits access to matching conditions. • For *network length*, specify the network number and length (in bits) of the network mask that is permitted or denied.
Step 4	**ip msdp description** {*peer-name* \| *peer-address*} *text*	(Optional) Configure a description for the specified peer to make it easier to identify in a configuration or in **show** command output. By default, no description is associated with an MSDP peer.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the default peer, use the **no ip msdp default-peer** *ip-address* | *name* global configuration command.

This example shows a partial configuration of Router A and Router C in Figure 35-2. Each of these ISPs have more than one customer (like the customer in Figure 35-2) who use default peering (no BGP or MBGP). In that case, they might have similar configurations. That is, they accept SAs only from a default peer if the SA is permitted by the corresponding prefix list.

Router A

```
Router(config)# ip msdp default-peer 10.1.1.1
Router(config)# ip msdp default-peer 10.1.1.1 prefix-list site-a
Router(config)# ip prefix-list site-b permit 10.0.0.0/1
```

Router C

```
Router(config)# ip msdp default-peer 10.1.1.1 prefix-list site-a
Router(config)# ip prefix-list site-b permit 10.0.0.0/1
```

Caching Source-Active State

By default, the switch does not cache source/group pairs from received SA messages. When the switch forwards the MSDP SA information, it does not store it in memory. Therefore, if a member joins a group soon after a SA message is received by the local RP, that member needs to wait until the next SA message to hear about the source. This delay is known as join latency.

If you want to sacrifice some memory in exchange for reducing the latency of the source information, you can configure the switch to cache SA messages.

Beginning in privileged EXEC mode, follow these steps to enable the caching of source/group pairs. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip msdp cache-sa-state** [**list** *access-list-number*]	Enable the caching of source/group pairs (create an SA state). Those pairs that pass the access list are cached. For **list** *access-list-number*, the range is 100 to 199.
Step 3	**access-list** *access-list-number* {**deny** \| **permit**} *protocol source source-wildcard destination destination-wildcard*	Create an IP extended access list, repeating the command as many times as necessary. • For *access-list-number*, the range is 100 to 199. Enter the same number created in Step 2. • The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. • For *protocol*, enter **ip** as the protocol name. • For *source*, enter the number of the network or host from which the packet is being sent. • For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. • For *destination*, enter the number of the network or host to which the packet is being sent. • For *destination-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the destination. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Note An alternative to this command is the **ip msdp sa-request** global configuration command, which causes the switch to send an SA request message to the MSDP peer when a new member for a group becomes active. For more information, see the next section.

To return to the default setting (no SA state is created), use the **no ip msdp cache-sa-state** global configuration command.

This example shows how to enable the cache state for all sources in 171.69.0.0/16 sending to groups 224.2.0.0/16:

```
Switch(config)# ip msdp cache-sa-state 100
Switch(config)# access-list 100 permit ip 171.69.0.0 0.0.255.255 224.2.0.0 0.0.255.255
```

Requesting Source Information from an MSDP Peer

Local RPs can send SA requests and get immediate responses for all active sources for a given group. By default, the switch does not send any SA request messages to its MSDP peers when a new member joins a group and wants to receive multicast traffic. The new member waits to receive the next periodic SA message.

If you want a new member of a group to learn the active multicast sources in a connected PIM sparse-mode domain that are sending to a group, configure the switch to send SA request messages to the specified MSDP peer when a new member joins a group. The peer replies with the information in its SA cache. If the peer does not have a cache configured, this command has no result. Configuring this feature reduces join latency but sacrifices memory.

Beginning in privileged EXEC mode, follow these steps to configure the switch to send SA request messages to the MSDP peer when a new member joins a group and wants to receive multicast traffic. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip msdp sa-request** {*ip-address* \| *name*}	Configure the switch to send SA request messages to the specified MSDP peer. For *ip-address* \| *name*, enter the IP address or name of the MSDP peer from which the local switch requests SA messages when a new member for a group becomes active. Repeat the command for each MSDP peer that you want to supply with SA messages.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no ip msdp sa-request** {*ip-address* | *name*} global configuration command.

This example shows how to configure the switch to send SA request messages to the MSDP peer at 171.69.1.1:

```
Switch(config)# ip msdp sa-request 171.69.1.1
```

Controlling Source Information that Your Switch Originates

You can control the multicast source information that originates with your switch:

- Sources you advertise (based on your sources)
- Receivers of source information (based on knowing the requestor)

For more information, see the "Redistributing Sources" section on page 35-9 and the "Filtering Source-Active Request Messages" section on page 35-11.

Redistributing Sources

SA messages originate on RPs to which sources have registered. By default, any source that registers with an RP is advertised. The *A flag* is set in the RP when a source is registered, which means the source is advertised in an SA unless it is filtered.

Beginning in privileged EXEC mode, follow these steps to further restrict which registered sources are advertised. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip msdp redistribute** [**list** *access-list-name*] [**asn** *aspath-access-list-number*] [**route-map** *map*]	Configure which (S,G) entries from the multicast routing table are advertised in SA messages. By default, only sources within the local domain are advertised. • (Optional) For **list** *access-list-name*, enter the name or number of an IP standard or extended access list. The range is 1 to 99 for standard access lists and 100 to 199 for extended lists. The access list controls which local sources are advertised and to which groups they send. • (Optional) For **asn** *aspath-access-list-number*, enter the IP standard or extended access list number in the range 1 to 199. This access list number must also be configured in the **ip as-path access-list** command. • (Optional) For **route-map** *map*, enter the IP standard or extended access list number in the range 1 to 199. This access list number must also be configured in the **ip as-path access-list** command. The switch advertises (S,G) pairs according to the access list or autonomous system path access list.

	Command	Purpose
Step 3	**access-list** *access-list-number* {**deny** \| **permit**} *source* [*source-wildcard*]	Create an IP standard access list, repeating the command as many times as necessary.
	or	or
	access-list *access-list-number* {**deny** \| **permit**} *protocol source source-wildcard destination destination-wildcard*	Create an IP extended access list, repeating the command as many times as necessary. • For *access-list-number*, the range is 1 to 99 for standard access lists and 100 to 199 for extended lists. Enter the same number created in Step 2. • The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. • For *protocol*, enter **ip** as the protocol name. • For *source*, enter the number of the network or host from which the packet is being sent. • For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. • For *destination*, enter the number of the network or host to which the packet is being sent. • For *destination-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the destination. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the filter, use the **no ip msdp redistribute** global configuration command.

Filtering Source-Active Request Messages

By default, only switches that are caching SA information can respond to SA requests. By default, such a switch honors all SA request messages from its MSDP peers and supplies the IP addresses of the active sources.

However, you can configure the switch to ignore all SA requests from an MSDP peer. You can also honor only those SA request messages from a peer for groups described by a standard access list. If the groups in the access list pass, SA request messages are accepted. All other such messages from the peer for other groups are ignored.

Beginning in privileged EXEC mode, follow these steps to configure one of these options. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip msdp filter-sa-request** *ip-address* \| *name* or **ip msdp filter-sa-request** {*ip-address* \| *name*} **list** *access-list-number*	Filter all SA request messages from the specified MSDP peer. or Filter SA request messages from the specified MSDP peer for groups that pass the standard access list. The access list describes a multicast group address. The range for the access-list-number is 1 to 99.
Step 3	**access-list** *access-list-number* {**deny** \| **permit**} *source* [*source-wildcard*]	Create an IP standard access list, repeating the command as many times as necessary. • For *access-list-number*, the range is 1 to 99. • The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. • For *source*, enter the number of the network or host from which the packet is being sent. • (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no ip msdp filter-sa-request** {*ip-address* | *name*} global configuration command.

This example shows how to configure the switch to filter SA request messages from the MSDP peer at 171.69.2.2. SA request messages from sources on network 192.4.22.0 pass access list 1 and are accepted; all others are ignored.

```
Switch(config)# ip msdp filter sa-request 171.69.2.2 list 1
Switch(config)# access-list 1 permit 192.4.22.0 0.0.0.255
```

Controlling Source Information that Your Switch Forwards

By default, the switch forwards all SA messages it receives to all its MSDP peers. However, you can prevent outgoing messages from being forwarded to a peer by using a filter or by setting a time-to-live (TTL) value. These methods are described in the next sections.

Using a Filter

By creating a filter, you can perform one of these actions:

- Filter all source/group pairs
- Specify an IP extended access list to pass only certain source/group pairs
- Filter based on match criteria in a route map

Beginning in privileged EXEC mode, follow these steps to apply a filter. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip msdp sa-filter out** *ip-address* \| *name*	Filter all SA messages to the specified MSDP peer.
	or	or
	ip msdp sa-filter out {*ip-address* \| *name*} **list** *access-list-number*	To the specified peer, pass only those SA messages that pass the IP extended access list. The range for the extended *access-list-number* is 100 to 199. If both the **list** and the **route-map** keywords are used, all conditions must be true to pass any (S,G) pair in outgoing SA messages.
	or	or
	ip msdp sa-filter out {*ip-address* \| *name*} **route-map** *map-tag*	To the specified MSDP peer, pass only those SA messages that meet the match criteria in the route map *map-tag*. If all match criteria are true, a **permit** from the route map passes routes through the filter. A **deny** filters routes.

	Command	Purpose
Step 3	**access-list** *access-list-number* {**deny** \| **permit**} *protocol source source-wildcard destination destination-wildcard*	(Optional) Create an IP extended access list, repeating the command as many times as necessary. • For *access-list-number*, enter the number specified in Step 2. • The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. • For *protocol*, enter **ip** as the protocol name. • For *source*, enter the number of the network or host from which the packet is being sent. • For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. • For *destination*, enter the number of the network or host to which the packet is being sent. • For *destination-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the destination. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the filter, use the **no ip msdp sa-filter out** {*ip-address* | *name*} [**list** *access-list-number*] [**route-map** *map-tag*] global configuration command.

This example shows how to allow only (S,G) pairs that pass access list 100 to be forwarded in an SA message to the peer named *switch.cisco.com*:

```
Switch(config)# ip msdp peer switch.cisco.com connect-source gigabitethernet0/1
Switch(config)# ip msdp sa-filter out switch.cisco.com list 100
Switch(config)# access-list 100 permit ip 171.69.0.0 0.0.255.255 224.20 0 0.0.255.255
```

Using TTL to Limit the Multicast Data Sent in SA Messages

You can use a TTL value to control what data is encapsulated in the first SA message for every source. Only multicast packets with an IP-header TTL greater than or equal to the *ttl* argument are sent to the specified MSDP peer. For example, you can limit internal traffic to a TTL of 8. If you want other groups to go to external locations, you must send those packets with a TTL greater than 8.

Beginning in privileged EXEC mode, follow these steps to establish a TTL threshold. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip msdp ttl-threshold** {*ip-address* \| *name*} *ttl*	Limit which multicast data is encapsulated in the first SA message to the specified MSDP peer. • For *ip-address* \| *name*, enter the IP address or name of the MSDP peer to which the TTL limitation applies. • For *ttl*, enter the TTL value. The default is 0, which means all multicast data packets are forwarded to the peer until the TTL is exhausted. The range is 0 to 255.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no ip msdp ttl-threshold** {*ip-address* | *name*} global configuration command.

Controlling Source Information that Your Switch Receives

By default, the switch receives all SA messages that its MSDP RPF peers send to it. However, you can control the source information that you receive from MSDP peers by filtering incoming SA messages. In other words, you can configure the switch to not accept them.

You can perform one of these actions:

- Filter all incoming SA messages from an MSDP peer
- Specify an IP extended access list to pass certain source/group pairs
- Filter based on match criteria in a route map

Beginning in privileged EXEC mode, follow these steps to apply a filter. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip msdp sa-filter in** *ip-address* \| *name*	Filter all SA messages from the specified MSDP peer.
	or	or
	ip msdp sa-filter in {*ip-address* \| *name*} **list** *access-list-number*	From the specified peer, pass only those SA messages that pass the IP extended access list. The range for the extended *access-list-number* is 100 to 199. If both the **list** and the **route-map** keywords are used, all conditions must be true to pass any (S,G) pair in incoming SA messages.
	or	or
	ip msdp sa-filter in {*ip-address* \| *name*} **route-map** *map-tag*	From the specified MSDP peer, pass only those SA messages that meet the match criteria in the route map *map-tag*. If all match criteria are true, a **permit** from the route map passes routes through the filter. A **deny** will filter routes.
Step 3	**access-list** *access-list-number* {**deny** \| **permit**} *protocol source source-wildcard destination destination-wildcard*	(Optional) Create an IP extended access list, repeating the command as many times as necessary. • For *access-list-number*, enter the number specified in Step 2. • The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. • For *protocol*, enter **ip** as the protocol name. • For *source*, enter the number of the network or host from which the packet is being sent. • For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. • For *destination*, enter the number of the network or host to which the packet is being sent. • For *destination-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the destination. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the filter, use the **no ip msdp sa-filter in** {*ip-address* | *name*} [**list** *access-list-number*] [**route-map** *map-tag*] global configuration command.

This example shows how to filter all SA messages from the peer named *switch.cisco.com*:

```
Switch(config)# ip msdp peer switch.cisco.com connect-source gigabitethernet0/1
Switch(config)# ip msdp sa-filter in switch.cisco.com
```

Configuring an MSDP Mesh Group

An MSDP mesh group is a group of MSDP speakers that have fully meshed MSDP connectivity among one another. Any SA messages received from a peer in a mesh group are not forwarded to other peers in the same mesh group. Thus, you reduce SA message flooding and simplify peer-RPF flooding. Use the **ip msdp mesh-group** global configuration command when there are multiple RPs within a domain. It is especially used to send SA messages across a domain. You can configure multiple mesh groups (with different names) in a single switch.

Beginning in privileged EXEC mode, follow these steps to create a mesh group. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip msdp mesh-group** *name* {*ip-address* \| *name*}	Configure an MSDP mesh group, and specify the MSDP peer belonging to that mesh group. By default, the MSDP peers do not belong to a mesh group. • For *name*, enter the name of the mesh group. • For *ip-address* \| *name*, enter the IP address or name of the MSDP peer to be a member of the mesh group.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.
Step 6		Repeat this procedure on each MSDP peer in the group.

To remove an MSDP peer from a mesh group, use the **no ip msdp mesh-group** *name* {*ip-address* | *name*} global configuration command.

Shutting Down an MSDP Peer

If you want to configure many MSDP commands for the same peer and you do not want the peer to become active, you can shut down the peer, configure it, and later bring it up. When a peer is shut down, the TCP connection is terminated and is not restarted. You can also shut down an MSDP session without losing configuration information for the peer.

Beginning in privileged EXEC mode, follow these steps to shut down a peer. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip msdp shutdown** {*peer-name* \| *peer address*}	Administratively shut down the specified MSDP peer without losing configuration information. For *peer-name* \| *peer address*, enter the IP address or name of the MSDP peer to shut down.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To bring the peer back up, use the **no ip msdp shutdown** {*peer-name* | *peer address*} global configuration command. The TCP connection is reestablished

Including a Bordering PIM Dense-Mode Region in MSDP

You can configure MSDP on a switch that borders a PIM sparse-mode region with a dense-mode region. By default, active sources in the dense-mode region do not participate in MSDP.

Note We do not recommend using the **ip msdp border sa-address** global configuration command. It is better to configure the border router in the sparse-mode domain to proxy-register sources in the dense-mode domain to the RP of the sparse-mode domain and have the sparse-mode domain use standard MSDP procedures to advertise these sources.

Beginning in privileged EXEC mode, follow these steps to configure the border router to send SA messages for sources active in the dense-mode region to the MSDP peers. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip msdp border sa-address** *interface-id*	Configure the switch on the border between a dense-mode and sparse-mode region to send SA messages about active sources in the dense-mode region. For *interface-id*, specify the interface from which the IP address is derived and used as the RP address in SA messages. The IP address of the interface is used as the Originator-ID, which is the RP field in the SA message.
Step 3	**ip msdp redistribute** [**list** *access-list-name*] [**asn** *aspath-access-list-number*] [**route-map** *map*]	Configure which (S,G) entries from the multicast routing table are advertised in SA messages. For more information, see the "Redistributing Sources" section on page 35-9.
Step 4	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Note that the **ip msdp originator-id** global configuration command also identifies an interface to be used as the RP address. If both the **ip msdp border sa-address** and the **ip msdp originator-id** global configuration commands are configured, the address derived from the **ip msdp originator-id** command specifies the RP address.

To return to the default setting (active sources in the dense-mode region do not participate in MSDP), use the **no ip msdp border sa-address** *interface-id* global configuration command.

Configuring an Originating Address other than the RP Address

You can allow an MSDP speaker that originates an SA message to use the IP address of the interface as the RP address in the SA message by changing the Originator ID. You might change the Originator ID in one of these cases:

- If you configure a logical RP on multiple switches in an MSDP mesh group.
- If you have a switch that borders a PIM sparse-mode domain and a dense-mode domain. If a switch borders a dense-mode domain for a site, and sparse-mode is being used externally, you might want dense-mode sources to be known to the outside world. Because this switch is not an RP, it would not have an RP address to use in an SA message. Therefore, this command provides the RP address by specifying the address of the interface.

Beginning in privileged EXEC mode, follow these steps to allow an MSDP speaker that originates an SA message to use the IP address on the interface as the RP address in the SA message. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip msdp originator-id** *interface-id*	Configures the RP address in SA messages to be the address of the originating device interface. For *interface-id*, specify the interface on the local switch.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

If both the **ip msdp border sa-address** and the **ip msdp originator-id** global configuration commands are configured, the address derived from the **ip msdp originator-id** command specifies the address of the RP.

To prevent the RP address from being derived in this way, use the **no ip msdp originator-id** *interface-id* global configuration command.

Monitoring and Maintaining MSDP

To monitor MSDP SA messages, peers, state, or peer status, use one or more of the privileged EXEC commands in Table 35-1:

Table 35-1 Commands for Monitoring and Maintaining MSDP

Command	Purpose
debug ip msdp [*peer-address* \| *name*] [**detail**] [**routes**]	Debugs an MSDP activity.
debug ip msdp resets	Debugs MSDP peer reset reasons.
show ip msdp count [*autonomous-system-number*]	Displays the number of sources and groups originated in SA messages from each autonomous system. The **ip msdp cache-sa-state** command must be configured for this command to produce any output.
show ip msdp peer [*peer-address* \| *name*]	Displays detailed information about an MSDP peer.
show ip msdp sa-cache [*group-address* \| *source-address* \| *group-name* \| *source-name*] [*autonomous-system-number*]	Displays (S,G) state learned from MSDP peers.
show ip msdp summary	Displays MSDP peer status and SA message counts.

To clear MSDP connections, statistics, or SA cache entries, use the privileged EXEC commands in Table 35-2:

Table 35-2 Commands for Clearing MSDP Connections, Statistics, or SA Cache Entries

Command	Purpose
clear ip msdp peer *peer-address* \| *name*	Clears the TCP connection to the specified MSDP peer, resetting all MSDP message counters.
clear ip msdp statistics [*peer-address* \| *name*]	Clears statistics counters for one or all the MSDP peers without resetting the sessions.
clear ip msdp sa-cache [*group-address* \| *name*]	Clears the SA cache entries for all entries, all sources for a specific group, or all entries for a specific source/group pair.



CHAPTER 36

Troubleshooting

This chapter describes how to identify and resolve software problems related to the Cisco IOS software on the Cisco ME 3400 switch.

You can use the command-line interface (CLI) to identify and solve problems.

Additional troubleshooting information related to hardware is provided in the hardware installation guide.

Note For complete syntax and usage information for the commands used in this chapter, see the command reference for this release and the *Cisco IOS Command Summary, Release 12.2*.

This chapter consists of these sections:

- Recovering from Corrupted Software By Using the Xmodem Protocol, page 36-2
- Recovering from a Lost or Forgotten Password, page 36-3

Note Recovery procedures require that you have physical access to the switch.

- Preventing Autonegotiation Mismatches, page 36-8
- SFP Module Security and Identification, page 36-9
- Monitoring SFP Module Status, page 36-9
- Using Ping, page 36-10
- Using Layer 2 Traceroute, page 36-13
- Using IP Traceroute, page 36-15
- Using TDR, page 36-17
- Using Debug Commands, page 36-18
- Using the show platform forward Command, page 36-19
- Using the crashinfo File, page 36-22

Recovering from Corrupted Software By Using the Xmodem Protocol

Switch software can be corrupted during an upgrade, by downloading the wrong file to the switch, and by deleting the image file. In all of these cases, the switch does not pass the power-on self-test (POST), and there is no connectivity.

This procedure uses the Xmodem Protocol to recover from a corrupt or wrong image file. There are many software packages that support the Xmodem Protocol, and this procedure is largely dependent on the emulation software that you are using.

This recovery procedure requires that you have physical access to the switch. The Cisco ME switch boot loader uses break-key detection to stop the automatic boot sequence for the password recovery purpose.

Note The break key character is different for each operating system.

On a SUN work station running UNIX, Ctrl-C is the break key.

On a PC running Windows XP or 2000, Ctrl-Break is the break key.

Cisco TAC has tabulated break keys for most common operating systems and an alternative *break key sequence* for those terminal emulators that do not support the break keys. See http://www.cisco.com/warp/public/701/61.html#how-to for that list.

Step 1 From your PC, download the software image tar file (*image_filename.tar*) from Cisco.com .

The Cisco IOS image is stored as a bin file in a directory in the tar file. For information about locating the software image files on Cisco.com, see the release notes

Step 2 Extract the bin file from the tar file.

- If you are using Windows, use a zip program that can read a tar file. Use the zip program to navigate to and extract the bin file.
- If you are using UNIX, follow these steps:

1. Display the contents of the tar file by using the **tar -tvf** <*image_filename.tar*> UNIX command.

```
switch% tar -tvf image_filename.tar
```

2. Locate the bin file, and extract it by using the **tar -xvf** <*image_filename.tar*> <*image_filename.bin*> UNIX command.

```
switch% tar -xvf image_filename.tar image_filename.bin
x me340x-metroipaccess-mz.122-25.EX.bin 3970586 bytes, 7756 tape blocks
```

3. Verify that the bin file was extracted by using the ls -l <*image_filename.bin*> UNIX command.

```
switch% ls -l image_filename.bin
-rw-r--r--   1 boba      3560586 Apr 21 12:00
me340x-metroipaccess-mz.122-25.EX.bin/
```

Step 3 Connect your PC with terminal-emulation software supporting the Xmodem Protocol to the switch console port.

Step 4 Set the line speed on the emulation software to 9600 baud.

Step 5 Unplug the switch power cord.

Step 6 Press the break key, and at the same time, reconnect the power cord to the switch.

You can release the break key a second or two after the LED above port 1 goes off. Several lines of information about the software appear along with instructions:

```
The system has been interrupted, or encountered an error during initializion of the flash
filesystem.  The following commands will initialize the flash filesystem, and finish
loading the operating system software:

flash_init
load_helper
boot
```

Step 7 Initialize the flash file system:

```
switch: flash_init
```

Step 8 If you had set the console port speed to anything other than 9600, it has been reset to that particular speed. Change the emulation software line speed to match that of the switch console port.

Step 9 Load any helper files:

```
switch: load_helper
```

Step 10 Start the file transfer by using the Xmodem Protocol.

```
switch: copy xmodem: flash:image_filename.bin
```

Step 11 After the Xmodem request appears, use the appropriate command on the terminal-emulation software to start the transfer and to copy the software image into flash memory.

Step 12 Boot the newly downloaded Cisco IOS image.

```
switch:boot flash:image_filename.bin
```

Step 13 Use the **archive download-sw** privileged EXEC command to download the software image to the switch.

Step 14 Use the **reload** privileged EXEC command to restart the switch and to verify that the new software image is operating properly.

Step 15 Delete the flash:*image_filename.bin* file from the switch.

Recovering from a Lost or Forgotten Password

The default configuration for the switch allows an end user to recover from a lost password by interrupting the boot process during power-on and by entering a new password.

Note On these switches, a system administrator can disable some of the functionality of password recovery by allowing an end user to reset a password only by agreeing to return to the default configuration. If you are an end user trying to reset a password when password recovery has been disabled, a status message shows this during the recovery process.

Disabling password recovery provides configuration file security by preventing unauthorized users from accessing the configuration file.

The Cisco ME switch boot loader uses break-key detection to stop the automatic boot sequence for the password recovery purpose.

Note The break key character is different for each operating system.

On a SUN work station running UNIX, Ctrl-C is the break key.

On a PC running Windows XP or 2000, Ctrl-Break is the break key.

Cisco TAC has tabulated break keys for most common operating systems and an alternative *break key sequence* for those terminal emulators that do not support the break keys. To see that list go to: http://www.cisco.com/warp/public/701/61.html#how-to

These sections describes how to recover a forgotten or lost switch password:

- Procedure with Password Recovery Enabled, page 36-5
- Procedure with Password Recovery Disabled, page 36-7

You enable or disable password recovery by using the **service password-recovery** global configuration command.

Follow the steps in this procedure if you have forgotten or lost the switch password.

Step 1 Connect a terminal or PC with terminal-emulation software to the switch console port.

Step 2 Set the line speed on the emulation software to 9600 baud.

Step 3 Power off the switch.

Reconnect the power cord to the switch.

Step 4 After the switch performs POST, the switch begins the autoboot process. The boot loader prompts the user for a break key character during the boot-up sequence, as shown in this example:

```
***** The system will autoboot in 5 seconds *****

Send a break key to prevent autobooting.
```

You must enter the break key on the console terminal within 5 seconds of receiving the message that the system will autoboot. The System LED flashes green until the break key is accepted. After the break key is accepted, the System LED turns off until after the switch boots.

Several lines of information about the software appear with instructions, informing you if the password recovery procedure has been disabled or not.

- If you see a message that begins with this:

  ```
  The system has been interrupted, or encountered an error during initializion of the
  flash filesystem.  The following commands will initialize the flash filesystem, and
  finish loading the operating system software:

  flash_init
  load_helper
  boot
  ```

 proceed to the "Procedure with Password Recovery Enabled" section on page 36-5, and follow the steps.

- If you see a message that begins with this:

  ```
  The password-recovery mechanism has been triggered, but is currently disabled.
  ```

 proceed to the "Procedure with Password Recovery Disabled" section on page 36-7, and follow the steps.

Step 5 After recovering the password, reload the switch:

```
Switch> reload s
Proceed with reload? [confirm] y
```

Procedure with Password Recovery Enabled

If the password-recovery mechanism is enabled, this message appears:

```
The system has been interrupted, or encountered an error during initializion of the flash
filesystem.  The following commands will initialize the flash filesystem, and finish
loading the operating system software:

flash_init
load_helper
boot
```

Step 1 Initialize the flash file system:

```
switch: flash_init
```

Step 2 If you had set the console port speed to anything other than 9600, it has been reset to that particular speed. Change the emulation software line speed to match that of the switch console port.

Step 3 Load any helper files:

```
switch: load_helper
```

Step 4 Display the contents of flash memory:

```
switch: dir flash:
```

The switch file system appears:

```
Directory of flash:
   13  drwx         192   Mar 01 1993 22:30:48 me340x-metroipaccess-mz.122-25.EX
   11  -rwx        5825   Mar 01 1993 22:31:59  config.text
   18  -rwx         720   Mar 01 1993 02:21:30  vlan.dat

16128000 bytes total (10003456 bytes free)
```

Step 5 Rename the configuration file to config.text.old.

This file contains the password definition.

```
switch: rename flash:config.text flash:config.text.old
```

Step 6 Boot the system:

```
switch: boot
```

You are prompted to start the setup program. Enter **N** at the prompt:

```
Continue with the configuration dialog? [yes/no]: N
```

Step 7 At the switch prompt, enter privileged EXEC mode:

```
Switch> enable
```

Step 8 Rename the configuration file to its original name:

```
Switch# rename flash:config.text.old flash:config.text
```

Step 9 Copy the configuration file into memory:

```
Switch# copy flash:config.text system:running-config
Source filename [config.text]?
Destination filename [running-config]?
```

Press **Return** in response to the confirmation prompts.

The configuration file is now reloaded, and you can change the password.

Step 10 Enter global configuration mode:

```
Switch# configure terminal
```

Step 11 Change the password:

```
Switch (config)# enable secret password
```

The secret password can be from 1 to 25 alphanumeric characters, can start with a number, is case sensitive, and allows spaces but ignores leading spaces.

Step 12 Return to privileged EXEC mode:

```
Switch (config)# exit
Switch#
```

Step 13 Write the running configuration to the startup configuration file:

```
Switch# copy running-config startup-config
```

The new password is now in the startup configuration.

Note This procedure is likely to leave your switch virtual interface in a shutdown state. You can identify this interface by entering the **show running-config** privileged EXEC command. To re-enable the interface, enter the **interface vlan** *vlan-id* global configuration command, and specify the VLAN ID of the shutdown interface. With the switch in interface configuration mode, enter the **no shutdown** command.

Step 14 Reload the switch:

```
Switch# reload
```

Procedure with Password Recovery Disabled

If the password-recovery mechanism is disabled, this message appears:

```
The password-recovery mechanism has been triggered, but
is currently disabled.  Access to the boot loader prompt
through the password-recovery mechanism is disallowed at
this point.  However, if you agree to let the system be
reset back to the default system configuration, access
to the boot loader prompt can still be allowed.

Would you like to reset the system back to the default configuration (y/n)?
```

Caution Returning the switch to the default configuration results in the loss of all existing configurations. We recommend that you contact your system administrator to verify if there are backup switch and VLAN configuration files.

Note Disabling password recovery provides configuration file security by preventing unauthorized users from accessing the configuration file.

- If you enter **n** (no), the normal boot process continues as if the break key had not been pressed; you cannot access the boot loader prompt, and you cannot enter a new password. You see the message:

  ```
  Press Enter to continue........
  ```

- If you enter **y** (yes), the configuration file in flash memory and the VLAN database file are deleted. When the default configuration loads, you can reset the password.

Step 1 Elect to continue with password recovery and lose the existing configuration:

```
Would you like to reset the system back to the default configuration (y/n)? Y
```

Step 2 Load any helper files:

```
Switch: load_helper
```

Step 3 Display the contents of flash memory:

```
switch: dir flash:
```

The switch file system appears:

```
Directory of flash:
13  drwx         192   Mar 01 1993 22:30:48 me340x-metroipaccess-mz.122-25.EX

16128000 bytes total (10003456 bytes free)
```

Step 4 Boot the system:

```
Switch: boot
```

You are prompted to start the setup program. To continue with password recovery, enter **N** at the prompt:

```
Continue with the configuration dialog? [yes/no]: N
```

Step 5 At the switch prompt, enter privileged EXEC mode:

```
Switch> enable
```

Step 6 Enter global configuration mode:

```
Switch# configure terminal
```

Step 7 Change the password:

```
Switch (config)# enable secret password
```

The secret password can be from 1 to 25 alphanumeric characters, can start with a number, is case sensitive, and allows spaces but ignores leading spaces.

Step 8 Return to privileged EXEC mode:

```
Switch (config)# exit
Switch#
```

Step 9 Write the running configuration to the startup configuration file:

```
Switch# copy running-config startup-config
```

The new password is now in the startup configuration.

Note This procedure is likely to leave your switch virtual interface in a shutdown state. You can identify this interface by entering the **show running-config** privileged EXEC command. To re-enable the interface, enter the **interface vlan** *vlan-id* global configuration command, and specify the VLAN ID of the shutdown interface. With the switch in interface configuration mode, enter the **no shutdown** command.

Step 10 You must now reconfigure the switch. If the system administrator has the backup switch and VLAN configuration files available, you should use those.

Preventing Autonegotiation Mismatches

The IEEE 802.3ab autonegotiation protocol manages the switch settings for speed (10, 100, and 1000 Mbps, excluding SFP module ports) and duplex (half or full). There are situations when this protocol can incorrectly align these settings, reducing performance. A mismatch occurs under these circumstances:

- A manually set speed or duplex parameter is different from the manually set speed or duplex parameter on the connected port.
- A port is set to autonegotiate, and the connected port is set to full duplex with no autonegotiation.

To maximize switch performance and ensure a link, follow one of these guidelines when changing the settings for duplex and speed:

- Let both ports autonegotiate both speed and duplex.
- Manually set the speed and duplex parameters for the ports on both ends of the connection.

Note If a remote device does not autonegotiate, configure the duplex settings on the two ports to match. The speed parameter can adjust itself even if the connected port does not autonegotiate.

SFP Module Security and Identification

Cisco small form-factor pluggable (SFP) modules have a serial EEPROM that contains the module serial number, the vendor name and ID, a unique security code, and cyclic redundancy check (CRC). When an SFP module is inserted in the switch, the switch software reads the EEPROM to verify the serial number, vendor name and vendor ID, and recompute the security code and CRC. If the serial number, the vendor name or vendor ID, the security code, or CRC is invalid, the software generates a security error message and places the interface in an error-disabled state.

Note The security error message references the GBIC_SECURITY facility. The switch supports SFP modules and does not support GBIC modules. Although the error message text refers to GBIC interfaces and modules, the security messages actually refer to the SFP modules and module interfaces. For more information about error messages, see the system message guide *f*or this release.

If you are using a non-Cisco SFP module, remove the SFP module from the switch, and replace it with a Cisco module. After inserting a Cisco SFP module, use the **errdisable recovery cause gbic-invalid** global configuration command to verify the port status, and enter a time interval for recovering from the error-disabled state. After the elapsed interval, the switch brings the interface out of the error-disabled state and retries the operation. For more information about the **errdisable recovery** command, see the command reference for this release.

If the module is identified as a Cisco SFP module, but the system is unable to read vendor-data information to verify its accuracy, an SFP module error message is generated. In this case, you should remove and re-insert the SFP module. If it continues to fail, the SFP module might be defective.

Monitoring SFP Module Status

You can check the physical or operational status of an SFP module by using the **show interfaces transceiver** privileged EXEC command. This command shows the operational status, such as the temperature and the current for an SFP module on a specific interface and the alarm status. You can also use the command to check the speed and the duplex settings on an SFP module. For more information, see the **show interfaces transceiver** command in the command reference for this release.

Monitoring Temperature

The Cisco ME switch monitors the temperature conditions. The switch also uses the temperature information to control the fans.

Use the **show env temperature** status privileged EXEC command to display the temperature value, state, and thresholds. The temperature value is the temperature in the switch (not the external temperature).You can configure only the yellow threshold level (in Celsius) by using the **system env temperature threshold yellow** *value* global configuration command to set the difference between the yellow and red thresholds. You cannot configure the green or red thresholds. For more information, see the command reference for this release.

Using Ping

These sections contain this information:

- Understanding Ping, page 36-10
- Using Ping, page 36-10

Understanding Ping

The Cisco ME switch supports IP ping, which you can use to test connectivity to remote hosts. Ping sends an echo request packet to an address and waits for a reply.

The Cisco ME switch also provides the Control Plane Security feature, which by default drops ping response packets received on user network interfaces (UNIs). However, methods are available to ping successfully from the switch to a host connected to a UNI.

Control Plane Security does not drop ping response packets to or from network node interfaces (NNIs), and no special configuration is required to enable pings to or from hosts connected to NNIs.

Using Ping

Beginning in privileged EXEC mode, use the **ping** command to ping another device on the network from the switch:

Command	Purpose
ping [*host* \| *address*]	Ping a remote host by supplying the hostname or IP network address. **Note** Though other protocol keywords are available with the **ping** command, they are not supported in this release.

Note Ping is not supported on a UNI configured as an IEEE 802.1Q tunnel port.

Ping is supported on NNIs on all software images.

It is important to note that three software images are available for the Cisco ME 3400 switch, which provide different options for pinging a host connected to a UNI:

- Metro IP access image that supports IP routing
- Metro access image
- Metro base image

The next sections apply to both access ports and trunk ports.

All Software Versions

For all software images for the Cisco ME switch, you can use a Layer 3 service policy to enable pings from the switch to a host connected to a UNI.

Note For a switch running the metro IP access image, IP routing is not enabled by default and does not have to be enabled to use a Layer 3 service policy.

This example is one possible configuration:

```
switch# configure terminal
switch(config)# access list 101 permit ip any any
switch(config)# class-map match-any ping-class
switch(config-cmap)# match access-group 101
switch(config-cmap)# exit
switch(config)# policy-map ping-policy
switch(config-pmap)# class ping-class
switch(config-pmap-c)# police 1000000
switch(config-pmap-c)# exit
switch(config-pmap)# exit
switch(config)# int fa0/1
switch(config-if)# service-policy input ping-policy
switch(config-if)# switchport access vlan 2
switch(config-if)# no shut
switch(config-if)# exit
switch(config)# int vlan 2
switch(config-if)# ip address 192.168.1.1 255.255.255.0
switch(config-if)# end
switch# ping 192.168.1.2
```

Metro IP Access Image

When your switch is running the the metro IP access image, you can use any of these methods:

- Apply a Layer 3 service policy to a UNI.
- Enable IP routing globally and ping from a switch virtual interface (SVI).
- Enable IP routing and ping from a routed port.

For a sample configuration of how to add a Layer 3 service policy to a UNI, see the "All Software Versions" section.

For examples using IP routing and pinging from an SVI or a routed port, see the next sections.

IP Routing and SVI

IP routing is only supported on UNIs when the switch is running the metro IP access image.

You can use this configuration to enable IP routing and enable pings from an SVI to a host connected to a UNI.

```
Switch# configure terminal
Switch(config)# ip routing
Switch(config)# int fa0/1
Switch(config-if)# switchport access vlan 2
Switch(config-if)# no shutdown
Switch(config-if)# int vlan 2
Switch(config-if)# ip address 192.168.1.1 255.255.255.0
Switch(config-if)# end

Switch# ping 192.168.1.2
```

With this configuration, a host with an IP address of 192.168.1.2 can be pinged from the switch.

IP Routing and Routed Port

You can use this configuration to enable IP routing, change a switchport to a routed port, and permit pings from the switch to a connected host:

```
switch# configure terminal
switch(config)# int fa0/1
switch(config-if)# no switchport
switch(config-if)# ip address 192.168.1.1 255.255.255.0
switch(config-if)# no shutdown
switch(config-if)# exit
switch(config)# ip routing
switch(config)# end
switch# ping 192.168.1.2
```

Ping Responses

This response is typical of a successful ping to a host:

```
Switch# ping 72.20.52.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echoes to 172.20.52.3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
Switch#
```

An unsuccessful ping results in this message:

```
Switch# ping 72.20.52.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echoes to 172.20.52.3, timeout is 2 seconds:
. . . . .

Success rate is 0 percent (0/5)
```

Summary

Keep these guidelines in mind while pinging:

- IP routing is available only with the metro IP access image and is disabled by default.
- To ping a host in a different IP subnetwork from the switch, you must have IP routing configured to route between the subnets, and a static route to the destination might also be appropriate. If you need to enable or configure IP routing, see Chapter 32, "Configuring IP Unicast Routing."
- All software versions can use a Layer 3 service policy to permit pings to and from a host connected to a UNI. For more information about policy maps, see the "Input and Output Policies" section on page 30-4.

If your switch is running the metro IP access image, use one of these methods to ping a host connected to a UNI:

- Use a Layer 3 service policy to permit pings to and from a host connected to a UNI.
- Enable global IP routing and configure a port as a routed port by using the **no switchport** interface configuration command.
- Enable global IP routing, create an SVI, and assign an IP address to it. For more information about SVIs, see the "Switch Virtual Interfaces" section on page 9-5.

To end a ping session, simultaneously press and release the **Ctrl**, **Shift**, and **6** keys, and then press the **X** key.

Using Layer 2 Traceroute

These sections contain this information:

- Understanding Layer 2 Traceroute, page 36-13
- Layer 2 Traceroute Usage Guidelines, page 36-14
- Displaying the Physical Path, page 36-15

Understanding Layer 2 Traceroute

The Layer 2 traceroute feature allows the switch to identify the physical path that a packet takes from a source device to a destination device. Layer 2 traceroute supports only unicast source and destination MAC addresses. It finds the path by using the MAC address tables of the switches in the path. When the switch detects a device in the path that does not support Layer 2 traceroute, the switch continues to send Layer 2 trace queries and lets them time out.

Note Layer 2 traceroute is available only on NNIs.

The switch can only identify the path from the source device to the destination device. It cannot identify the path that a packet takes from source host to the source device or from the destination device to the destination host.

Layer 2 Traceroute Usage Guidelines

These are the Layer 2 traceroute usage guidelines:

- Cisco Discovery Protocol (CDP) must be enabled on all the devices in the network. For Layer 2 traceroute to function properly, do not disable CDP.

Note CDP is available only on NNIs.

For a list of switches that support Layer 2 traceroute, see the "Layer 2 Traceroute Usage Guidelines" section on page 36-14. If any devices in the physical path are transparent to CDP, the switch cannot identify the path through these devices. For more information about enabling CDP, see Chapter 22, "Configuring CDP."

- A switch is reachable from another switch when you can test connectivity by using the **ping** privileged EXEC command. All switches in the physical path must be reachable from each other.
- The maximum number of hops identified in the path is ten.
- You can enter the **traceroute mac** or the **traceroute mac ip** privileged EXEC command on a switch that is not in the physical path from the source device to the destination device. All switches in the path must be reachable from this switch.
- The **traceroute mac** command output shows the Layer 2 path only when the specified source and destination MAC addresses belong to the same VLAN. If you specify source and destination MAC addresses that belong to different VLANs, the Layer 2 path is not identified, and an error message appears.
- If you specify a multicast source or destination MAC address, the path is not identified, and an error message appears.
- If the source or destination MAC address belongs to multiple VLANs, you must specify the VLAN to which both the source and destination MAC addresses belong. If the VLAN is not specified, the path is not identified, and an error message appears.
- The **traceroute mac ip** command output shows the Layer 2 path when the specified source and destination IP addresses belong to the same subnet. When you specify the IP addresses, the switch uses the Address Resolution Protocol (ARP) to associate the IP addresses with the corresponding MAC addresses and the VLAN IDs.
 - If an ARP entry exists for the specified IP address, the switch uses the associated MAC address and identifies the physical path.
 - If an ARP entry does not exist, the switch sends an ARP query and tries to resolve the IP address. If the IP address is not resolved, the path is not identified, and an error message appears.
- When multiple devices are attached to one port through hubs (for example, multiple CDP neighbors are detected on a port), the Layer 2 traceroute feature is not supported. When more than one CDP neighbor is detected on a port, the Layer 2 path is not identified, and an error message appears.
- This feature is not supported in Token Ring VLANs.

Displaying the Physical Path

You can display the physical path that a packet takes from a source device to a destination device by using one of these privileged EXEC commands:

- **tracetroute mac** [**interface** *interface-id*] {*source-mac-address*} [**interface** *interface-id*] {*destination-mac-address*} [**vlan** *vlan-id*] [**detail**]
- **tracetroute mac ip** {*source-ip-address* | *source-hostname*}{*destination-ip-address* | *destination-hostname*} [**detail**]

Note Layer 2 traceroute is available only on NNIs.

For more information, see the command reference for this release.

Using IP Traceroute

These sections contain this information:

- Understanding IP Traceroute, page 36-15
- Executing IP Traceroute, page 36-16

Understanding IP Traceroute

You can use IP traceroute to identify the path that packets take through the network on a hop-by-hop basis. The command output displays all network layer (Layer 3) devices, such as routers, that the traffic passes through on the way to the destination.

Your switches can participate as the source or destination of the **traceroute** privileged EXEC command and might or might not appear as a hop in the **traceroute** command output. If the switch is the destination of the traceroute, it is displayed as the final destination in the output. Intermediate switches do not show up in the output if they are only bridging the packet from one port to another within the same VLAN. However, if the intermediate switch is a multilayer switch that is routing a particular packet, this switch shows up as a hop in the output.

The **traceroute** privileged EXEC command uses the Time To Live (TTL) field in the IP header to cause routers and servers to generate specific return messages. Traceroute starts by sending a User Datagram Protocol (UDP) datagram to the destination host with the TTL field set to 1. If a router finds a TTL value of 1 or 0, it drops the datagram and sends an Internet Control Message Protocol (ICMP) time-to-live-exceeded message to the sender. Traceroute finds the address of the first hop by examining the source address field of this message.

To identify the next hop, traceroute sends a UDP packet with a TTL value of 2. The first router decrements the TTL field by 1 and sends the datagram to the next router. The second router sees a TTL value of 1, discards the datagram, and returns the time-to-live-exceeded message to the source. This process continues until the TTL is incremented to a value large enough for the datagram to reach the destination host (or until the maximum TTL is reached).

To learn when a datagram reaches its destination, traceroute sets the UDP destination port number in the datagram to a very large value that the destination host is unlikely to be using. When a host receives a datagram destined to itself containing a destination port number that is unused locally, it sends an ICMP

port-unreachable error to the source. Because all errors except port-unreachable errors come from intermediate hops, the receipt of a port-unreachable error means that this message was sent by the destination port.

Executing IP Traceroute

Beginning in privileged EXEC mode, follow this step to trace that the path packets take through the network:

Command	Purpose
traceroute ip *host*	Trace the path that packets take through the network.

Note Though other protocol keywords are available with the **traceroute** privileged EXEC command, they are not supported in this release.

This example shows how to perform a **traceroute** to an IP host:

```
Switch# traceroute ip 171.9.15.10

Type escape sequence to abort.
Tracing the route to 171.69.115.10

  1 172.2.52.1 0 msec 0 msec 4 msec
  2 172.2.1.203 12 msec 8 msec 0 msec
  3 171.9.16.6 4 msec 0 msec 0 msec
  4 171.9.4.5 0 msec 4 msec 0 msec
  5 171.9.121.34 0 msec 4 msec 4 msec
  6 171.9.15.9 120 msec 132 msec 128 msec
  7 171.9.15.10 132 msec 128 msec 128 msec
Switch#
```

The display shows the hop count, IP address of the router, and the round-trip time in milliseconds for each of the three probes that are sent.

Table 36-1 Traceroute Output Display Characters

Character	Description
*	The probe timed out.
?	Unknown packet type.
A	Administratively unreachable. Usually, this output means that an access list is blocking traffic.
H	Host unreachable.
N	Network unreachable.
P	Protocol unreachable.
Q	Source quench.
U	Port unreachable.

To end a trace in progress, enter the escape sequence (**Ctrl-^ X** by default). Simultaneously press and release the **Ctrl**, **Shift**, and **6** keys, and then press the **X** key.

Using TDR

These sections contain this information:

- Understanding TDR, page 36-17
- Running TDR and Displaying the Results, page 36-17

Understanding TDR

You can use the Time Domain Reflector (TDR) feature to diagnose and resolve cabling problems. When running TDR, a local device sends a signal through a cable and compares the reflected signal to the initial signal.

TDR is supported only on the copper Ethernet 10/100 ports on the Cisco ME switch.

TDR can detect these cabling problems:

- Open, broken, or cut twisted-pair wires—The wires are not connected to the wires from the remote device.
- Shorted twisted-pair wires—The wires are touching each other or the wires from the remote device. For example, a shorted twisted pair can occur if one wire of the twisted pair is soldered to the other wire.

If one of the twisted-pair wires is open, TDR can find the length at which the wire is open.

Use TDR to diagnose and resolve cabling problems in these situations:

- Replacing a switch
- Setting up a wiring closet
- Troubleshooting a connection between two devices when a link cannot be established or when it is not operating properly

Running TDR and Displaying the Results

To run TDR, enter the **test cable-diagnostics tdr interface** *interface-id* privileged EXEC command:

To display the results, enter the **show cable-diagnostics tdr interface** *interface-id* privileged EXEC command. For a description of the fields in the display, see the command reference for this release.

Note TDR is supported only on the copper Ethernet 10/100 ports on the Cisco ME switch.

Using Debug Commands

These sections explains how you use **debug** commands to diagnose and resolve internetworking problems:

- Enabling Debugging on a Specific Feature, page 36-18
- Enabling All-System Diagnostics, page 36-19
- Redirecting Debug and Error Message Output, page 36-19

Caution Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. It is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Note For complete syntax and usage information for specific **debug** commands, see the command reference for this release.

Enabling Debugging on a Specific Feature

All **debug** commands are entered in privileged EXEC mode, and most **debug** commands take no arguments. For example, beginning in privileged EXEC mode, enter this command to enable the debugging for Switched Port Analyzer (SPAN):

```
Switch# debug span-session
```

The switch continues to generate output until you enter the **no** form of the command.

If you enable a **debug** command and no output appears, consider these possibilities:

- The switch might not be properly configured to generate the type of traffic that you want to monitor. Use the **show running-config** command to check its configuration.
- Even if the switch is properly configured, it might not generate the type of traffic that you want to monitor during the particular period that debugging is enabled. Depending on the feature you are debugging, you can use commands such as the TCP/IP **ping** command to generate network traffic.

To disable debugging of SPAN, enter this command in privileged EXEC mode:

```
Switch# no debug span-session
```

Alternately, in privileged EXEC mode, you can enter the **undebug** form of the command:

```
Switch# undebug span-session
```

To display the state of each debugging option, enter this command in privileged EXEC mode:

```
Switch# show debugging
```

Enabling All-System Diagnostics

Beginning in privileged EXEC mode, enter this command to enable all-system diagnostics:

```
Switch# debug all
```

Caution Because debugging output takes priority over other network traffic, and because the **debug all** privileged EXEC command generates more output than any other **debug** command, it can severely diminish switch performance or even render it unusable. In virtually all cases, it is best to use more specific **debug** commands.

The **no debug all** privileged EXEC command disables all diagnostic output. Using the **no debug all** command is a convenient way to ensure that you have not accidentally left any **debug** commands enabled.

Redirecting Debug and Error Message Output

By default, the network server sends the output from **debug** commands and system error messages to the console. If you use this default, you can use a virtual terminal connection to monitor debug output instead of connecting to the console port.

Possible destinations include the console, virtual terminals, internal buffer, and UNIX hosts running a syslog server. The syslog format is compatible with 4.3 Berkeley Standard Distribution (BSD) UNIX and its derivatives.

Note Be aware that the debugging destination you use affects system overhead. Logging messages to the console produces very high overhead, whereas logging messages to a virtual terminal produces less overhead. Logging messages to a syslog server produces even less, and logging to an internal buffer produces the least overhead of any method.

For more information about system message logging, see Chapter 26, “Configuring System Message Logging.”

Using the show platform forward Command

The output from the **show platform forward** privileged EXEC command provides some useful information about the forwarding results if a packet entering an interface is sent through the system. Depending upon the parameters entered about the packet, the output provides lookup table results and port maps used to calculate forwarding destinations, bitmaps, and egress information.

Note For more syntax and usage information for the **show platform forward** command, see the switch command reference for this release.

Most of the information in the output from the command is useful mainly for technical support personnel, who have access to detailed information about the switch ASICs. However, packet forwarding information can also be helpful in troubleshooting.

This is an example of the output from the **show platform forward** command on Gigabit Ethernet port 1 in VLAN 5 when the packet entering that port is addressed to unknown MAC addresses. The packet should be flooded to all other ports in VLAN 5.

```
Switch# show platform forward gigabitethernet0/1 vlan 5 1.1.1 2.2.2 ip 13.1.1.1 13.2.2.2
udp 10 20
Global Port Number:24, Asic Number:5
Src Real Vlan Id:5, Mapped Vlan Id:5

Ingress:
 Lookup                   Key-Used                    Index-Hit  A-Data
InptACL  40_0D020202_0D010101-00_40000014_000A0000     01FFA   03000000
L2Local  80_00050002_00020002-00_00000000_00000000     00C71   0000002B
Station Descriptor:02340000, DestIndex:0239, RewriteIndex:F005

========================================
Egress:Asic 2, switch 1
Output Packets:

----------------------------------------
Packet 1
 Lookup                   Key-Used                    Index-Hit  A-Data
OutptACL 50_0D020202_0D010101-00_40000014_000A0000     01FFE   03000000

Port       Vlan      SrcMac          DstMac    Cos  Dscpv
Gi0/1    0005 0001.0001.0001  0002.0002.0002

----------------------------------------
Packet 2
 Lookup                   Key-Used                    Index-Hit  A-Data
OutptACL 50_0D020202_0D010101-00_40000014_000A0000     01FFE   03000000

Port       Vlan      SrcMac          DstMac    Cos  Dscpv
Gi0/2    0005 0001.0001.0001  0002.0002.0002

----------------------------------------
<output truncated>
----------------------------------------
Packet 10
 Lookup                   Key-Used                    Index-Hit  A-Data
OutptACL 50_0D020202_0D010101-00_40000014_000A0000     01FFE   03000000
Packet dropped due to failed DEJA_VU Check on Gi0/2
```

This is an example of the output when the packet coming in on Gigabit Ethernet port 1 in VLAN 5 is sent to an address already learned on the VLAN on another port. It should be forwarded from the port on which the address was learned.

```
Switch# show platform forward gigabitethernet0/1 vlan 5 1.1.1 0009.43a8.0145 ip 13.1.1.1
13.2.2.2 udp 10 20
Global Port Number:24, Asic Number:5
Src Real Vlan Id:5, Mapped Vlan Id:5

Ingress:
 Lookup                   Key-Used                    Index-Hit  A-Data
InptACL  40_0D020202_0D010101-00_40000014_000A0000     01FFA   03000000
L2Local  80_00050009_43A80145-00_00000000_00000000     00086   02010197
Station Descriptor:F0050003, DestIndex:F005, RewriteIndex:0003

========================================
Egress:Asic 3, switch 1
Output Packets:
```

```
----------------------------------------
Packet 1
 Lookup                  Key-Used                    Index-Hit  A-Data
OutptACL 50_0D020202_0D010101-00_40000014_000A0000     01FFE   03000000

Port       Vlan      SrcMac          DstMac    Cos  Dscpv
Gi0/2    0005 0001.0001.0001  0009.43A8.0145
```

This is an example of the output when the packet coming in on Gigabit Ethernet port 1 in VLAN 5 has a destination MAC address set to the router MAC address in VLAN 5 and the destination IP address unknown. Because there is no default route set, the packet should be dropped.

```
Switch# show platform forward gigabitethernet0/1 vlan 5 1.1.1 03.e319.ee44 ip 13.1.1.1
13.2.2.2 udp 10 20
Global Port Number:24, Asic Number:5
Src Real Vlan Id:5, Mapped Vlan Id:5

Ingress:
 Lookup                  Key-Used                    Index-Hit  A-Data
InptACL  40_0D020202_0D010101-00_41000014_000A0000     01FFA   03000000
L3Local  00_00000000_00000000-90_00001400_0D020202     010F0   01880290
L3Scndr  12_0D020202_0D010101-00_40000014_000A0000     034E0   000C001D_00000000
Lookup Used:Secondary
Station Descriptor:02260000, DestIndex:0226, RewriteIndex:0000
```

This is an example of the output when the packet coming in on Gigabit Ethernet port 1 in VLAN 5 has a destination MAC address set to the router MAC address in VLAN 5 and the destination IP address set to an IP address that is in the IP routing table. It should be forwarded as specified in the routing table.

```
Switch# show platform forward gigabitethernet0/1 vlan 5 1.1.1 03.e319.ee44 ip 110.1.5.5
16.1.10.5
Global Port Number:24, Asic Number:5
Src Real Vlan Id:5, Mapped Vlan Id:5

Ingress:
 Lookup                  Key-Used                    Index-Hit  A-Data
InptACL  40_10010A05_0A010505-00_41000014_000A0000     01FFA   03000000
L3Local  00_00000000_00000000-90_00001400_10010A05     010F0   01880290
L3Scndr  12_10010A05_0A010505-00_40000014_000A0000     01D28   30090001_00000000
Lookup Used:Secondary
Station Descriptor:F0070007, DestIndex:F007, RewriteIndex:0007

========================================
Egress:Asic 3, switch 1
Output Packets:

----------------------------------------
Packet 1
 Lookup                  Key-Used                    Index-Hit  A-Data
OutptACL 50_10010A05_0A010505-00_40000014_000A0000     01FFE   03000000

Port       Vlan      SrcMac          DstMac    Cos  Dscpv
Gi0/2    0007 XXXX.XXXX.0246  0009.43A8.0147
```

Using the crashinfo File

The crashinfo file saves information that helps Cisco technical support representatives to debug problems that caused the Cisco IOS image to fail (crash). The switch writes the crash information to the console at the time of the failure, and the file is created the next time you boot the Cisco IOS image after the failure (instead of while the system is failing).

The information in the file includes the Cisco IOS image name and version that failed, a list of the processor registers, and a stack trace. You can provide this information to the Cisco technical support representative by using the **show tech-support** privileged EXEC command.

All crashinfo files are kept in this directory on the flash file system:

flash:/crashinfo/crashinfo_*n* where *n* is a sequence number.

Each new crashinfo file that is created uses a sequence number that is larger than any previously existing sequence number, so the file with the largest sequence number describes the most recent failure. Version numbers are used instead of a timestamp because the switches do not include a real-time clock. You cannot change the name of the file that the system will use when it creates the file. However, after the file is created, you can use the **rename** privileged EXEC command to rename it, but the contents of the renamed file will not be displayed by the **show tech-support** privileged EXEC command. You can delete crashinfo files by using the **delete** privileged EXEC command.

You can display the most recent crashinfo file (that is, the file with the highest sequence number at the end of its filename) by entering the **show tech-support** privileged EXEC command. You also can access the file by using any command that can copy or display files, such as the **more** or the **copy** privileged EXEC command.



Supported MIBs

This appendix lists the supported management information base (MIBs) for this release on the Cisco ME 3400 Ethernet Access switch. It contains these sections:

- MIB List, page A-1
- Using FTP to Access the MIB Files, page A-3

MIB List

- BRIDGE-MIB (RFC1493)

Note The BRIDGE-MIB supports the context of a single VLAN. By default, SNMP messages using the configured community string always provide information for VLAN 1. To obtain the BRIDGE-MIB information for other VLANs, for example VLAN x, use this community string in the SNMP message: configured community string @x.

- CISCO-CDP-MIB
- CISCO-CONFIG-COPY-MIB
- CISCO-CONFIG-MAN-MIB
- CISCO-ENTITY-FRU-CONTROL-MIB
- CISCO-ENTITY-VENDORTYPE-OID-MIB
- CISCO-ENVMON-MIB
- CISCO-ETHERNET-ACCESS-MIB
- CISCO-FLASH-MIB (Flash memory on all switches is modeled as removable flash memory.)
- CISCO-FTP-CLIENT-MIB
- CISCO-HSRP-MIB

Note Layer 3 MIBs are available only when the metro IP access image is running on the switch.

- CISCO-HSRP-EXT-MIB (partial support)
- CISCO-IGMP-FILTER-MIB
- CISCO-IMAGE-MIB

- CISCO-L2L3-INTERFACE-CONFIG-MIB
- CISCO-LAG-MIB
- CISCO-MAC-NOTIFICATION-MIB
- CISCO-MEMORY-POOL-MIB
- CISCO-PAE-MIB
- CISCO-PAGP-MIB
- CISCO-PING-MIB
- CISCO-PORT-QOS-MIB
- CISCO-PRODUCTS-MIB
- CISCO-PROCESS-MIB
- CISCO-RTTMON-MIB
- CISCO-SMI-MIB
- CISCO-STACKMAKER-MIB
- CISCO-STP-EXTENSIONS-MIB
- CISCO-SYSLOG-MIB
- CISCO-TC-MIB
- CISCO-TCP-MIB
- CISCO-UDLDP-MIB
- CISCO-VLAN-IFTABLE-RELATIONSHIP-MIB
- CISCO-VLAN-MEMBERSHIP-MIB
- CISCO-VTP-MIB
- ENTITY-MIB
- ETHERLIKE-MIB
- IEEE8021-PAE-MIB
- IEEE8023-LAG-MIB
- IF-MIB (In and out counters for VLANs are not supported.)
- IGMP-MIB
- INET-ADDRESS-MIB
- IPMROUTE-MIB
- OLD-CISCO-CHASSIS-MIB
- OLD-CISCO-FLASH-MIB
- OLD-CISCO-INTERFACES-MIB
- OLD-CISCO-IP-MIB
- OLD-CISCO-SYS-MIB
- OLD-CISCO-TCP-MIB
- OLD-CISCO-TS-MIB
- PIM-MIB

- RFC1213-MIB (Functionality is as per the agent capabilities specified in the CISCO-RFC1213-CAPABILITY.my.)
- RFC1253-MIB (OSPF-MIB)
- RMON-MIB
- RMON2-MIB
- SNMP-FRAMEWORK-MIB
- SNMP-MPD-MIB
- SNMP-NOTIFICATION-MIB
- SNMP-TARGET-MIB
- SNMPv2-MIB
- TCP-MIB
- UDP-MIB

Note You can also use this URL for a list of supported MIBs for the Cisco ME switch: ftp://nm-tac.cisco.com/pub/mib_repo/
You can access other information about MIBs and Cisco products on the Cisco web site: http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

Using FTP to Access the MIB Files

You can obtain each MIB file by using this procedure:

Step 1 Make sure that your FTP client is in passive mode.

Note Some FTP clients do not support passive mode.

Step 2 Use FTP to access the server **ftp.cisco.com**.

Step 3 Log in with the username **anonymous**.

Step 4 Enter your e-mail username when prompted for the password.

Step 5 At the `ftp>` prompt, change directories to **/pub/mibs/v1** and **/pub/mibs/v2**.

Step 6 Use the **get** *MIB_filename* command to obtain a copy of the MIB file.



Working with the Cisco IOS File System, Configuration Files, and Software Images

This appendix describes how to manipulate the Cisco ME 3400 Ethernet Access switch flash file system, how to copy configuration files, and how to archive (upload and download) software images to a switch.

Note For complete syntax and usage information for the commands used in this chapter, see the switch command reference for this release and the *Cisco IOS Configuration Fundamentals Command Reference, Release 12.2*.

This appendix consists of these sections:

- Working with the Flash File System, page B-1
- Working with Configuration Files, page B-8
- Working with Software Images, page B-18

Working with the Flash File System

The flash file system is a single flash device on which you can store files. It also provides several commands to help you manage software image and configuration files. The default flash file system on the switch is named *flash:*.

These sections contain this configuration information:

- Displaying Available File Systems, page B-2
- Setting the Default File System, page B-3
- Displaying Information about Files on a File System, page B-3
- Creating and Removing Directories, page B-4
- Copying Files, page B-4
- Deleting Files, page B-5
- Creating, Displaying, and Extracting tar Files, page B-5
- Displaying the Contents of a File, page B-7

Displaying Available File Systems

To display the available file systems on your switch, use the **show file systems** privileged EXEC command as shown in this example.

```
Switch# show file systems
File Systems:
     Size(b)     Free(b)      Type  Flags   Prefixes
*   15998976     5135872     flash     rw   flash:
           -           -    opaque     rw   bs:
           -           -    opaque     rw   vb:
      524288      520138     nvram     rw   nvram:
           -           -   network     rw   tftp:
           -           -    opaque     rw   null:
           -           -    opaque     rw   system:
           -           -    opaque     ro   xmodem:
           -           -    opaque     ro   ymodem:
```

Table B-1 show file systems Field Descriptions

Field	Value
Size(b)	Amount of memory in the file system in bytes.
Free(b)	Amount of free memory in the file system in bytes.
Type	Type of file system. **flash**—The file system is for a flash memory device. **nvram**—The file system is for a NVRAM device. **opaque**—The file system is a locally generated *pseudo* file system (for example, the *system*) or a download interface, such as brimux. **unknown**—The file system is an unknown type.
Flags	Permission for file system. **ro**—read-only. **rw**—read/write.\ **wo**—write-only.
Prefixes	Alias for file system. **flash:**—Flash file system. **nvram:**—NVRAM. **null:**—Null destination for copies. You can copy a remote file to null to find its size. **rcp:**—Remote Copy Protocol (RCP) network server. **system:**—Contains the system memory, including the running configuration. **tftp:**—TFTP network server. **xmodem:**—Obtain the file from a network machine by using the Xmodem protocol. **ymodem:**—Obtain the file from a network machine by using the Ymodem protocol.

Setting the Default File System

You can specify the file system or directory that the system uses as the default file system by using the **cd** *filesystem:* privileged EXEC command. You can set the default file system to omit the *filesystem:* argument from related commands. For example, for all privileged EXEC commands that have the optional *filesystem:* argument, the system uses the file system specified by the **cd** command.

By default, the default file system is *flash:*.

You can display the current default file system as specified by the **cd** command by using the **pwd** privileged EXEC command.

Displaying Information about Files on a File System

You can view a list of the contents of a file system before manipulating its contents. For example, before copying a new configuration file to flash memory, you might want to verify that the file system does not already contain a configuration file with the same name. Similarly, before copying a flash configuration file to another location, you might want to verify its filename for use in another command.

To display information about files on a file system, use one of the privileged EXEC commands in Table B-2:

Table B-2 Commands for Displaying Information About Files

Command	Description
dir [/**all**] [*filesystem***:**][*filename*]	Display a list of files on a file system.
show file systems	Display more information about each of the files on a file system.
show file information *file-url*	Display information about a specific file.
show file descriptors	Display a list of open file descriptors. File descriptors are the internal representations of open files. You can use this command to see if another user has a file open.

Changing Directories and Displaying the Working Directory

Beginning in privileged EXEC mode, follow these steps to change directories and display the working directory.

	Command	Purpose
Step 1	**dir** *filesystem***:**	Display the directories on the specified file system. For *filesystem***:**, use **flash:** for the system board flash device.
Step 2	**cd new_configs**	Change to the directory of interest. The command example shows how to change to the directory named *new_configs*.
Step 3	**pwd**	Display the working directory.

Creating and Removing Directories

Beginning in privileged EXEC mode, follow these steps to create and remove a directory:

	Command	Purpose
Step 1	**dir** *filesystem***:**	Display the directories on the specified file system. For *filesystem***:**, use **flash:** for the system board flash device.
Step 2	**mkdir old_configs**	Create a new directory. The command example shows how to create the directory named *old_configs*. Directory names are case sensitive. Directory names are limited to 45 characters between the slashes (/); the name cannot contain control characters, spaces, deletes, slashes, quotes, semicolons, or colons.
Step 3	**dir** *filesystem***:**	Verify your entry.

To delete a directory with all its files and subdirectories, use the **delete /force /recursive** *filesystem***:/***file-url* privileged EXEC command.

Use the **/recursive** keyword to delete the named directory and all subdirectories and the files contained in it. Use the **/force** keyword to suppress the prompting that confirms a deletion of each file in the directory. You are prompted only once at the beginning of this deletion process. Use the **/force** and **/recursive** keywords for deleting old software images that were installed by using the **archive download-sw** command but are no longer needed.

For *filesystem*, use **flash:** for the system board flash device. For *file-url*, enter the name of the directory to be deleted. All the files in the directory and the directory are removed.

Caution When files and directories are deleted, their contents cannot be recovered.

Copying Files

To copy a file from a source to a destination, use the **copy** *source-url destination-url* privileged EXEC command. For the source and destination URLs, you can use **running-config** and **startup-config** keyword shortcuts. For example, the **copy running-config startup-config** command saves the currently running configuration file to the NVRAM section of flash memory to be used as the configuration during system initialization.

You can also copy from special file systems (**xmodem:**, **ymodem:**) as the source for the file from a network machine that uses the Xmodem or Ymodem protocol.

Network file system URLs include **ftp:**, **rcp:**, and **tftp:** and have these syntaxes:

- FTP—**ftp:**[[*//username* [**:***password*]**@***location*]*/directory*]*/filename*
- RCP—**rcp:**[[*//username***@***location*]*/directory*]*/filename*
- TFTP—**tftp:**[[*//location*]*/directory*]*/filename*

Local writable file systems include flash:.

Some invalid combinations of source and destination exist. Specifically, you cannot copy these combinations:

- From a running configuration to a running configuration
- From a startup configuration to a startup configuration
- From a device to the same device (for example, the **copy flash: flash:** command is invalid)

For specific examples of using the **copy** command with configuration files, see the "Working with Configuration Files" section on page B-8.

To copy software images either by downloading a new version or by uploading the existing one, use the **archive download-sw** or the **archive upload-sw** privileged EXEC command. For more information, see the "Working with Software Images" section on page B-18.

Deleting Files

When you no longer need a file on a flash memory device, you can permanently delete it. To delete a file or directory from a specified flash device, use the **delete** [**/force**] [**/recursive**] [*filesystem***:**]*/file-url* privileged EXEC command.

Use the **/recursive** keyword for deleting a directory and all subdirectories and the files contained in it. Use the **/force** keyword to suppress the prompting that confirms a deletion of each file in the directory. You are prompted only once at the beginning of this deletion process. Use the **/force** and **/recursive** keywords for deleting old software images that were installed by using the **archive download-sw** command but are no longer needed.

If you omit the *filesystem***:** option, the switch uses the default device specified by the **cd** command. For *file-url*, you specify the path (directory) and the name of the file to be deleted.

When you attempt to delete any files, the system prompts you to confirm the deletion.

Caution When files are deleted, their contents cannot be recovered.

This example shows how to delete the file *myconfig* from the default flash memory device:

```
Switch# delete myconfig
```

Creating, Displaying, and Extracting tar Files

You can create a tar file and write files into it, list the files in a tar file, and extract the files from a tar file as described in the next sections.

Note Instead of using the **copy** privileged EXEC command or the **archive tar** privileged EXEC command, we recommend using the **archive download-sw** and **archive upload-sw** privileged EXEC commands to download and upload software image files.

Creating a tar File

To create a tar file and write files into it, use this privileged EXEC command:

archive tar /create *destination-url* **flash:***/file-url*

For *destination-url*, specify the destination URL alias for the local or network file system and the name of the tar file to create. These options are supported:

- For the local flash file system, the syntax is **flash:**
- For the FTP, the syntax is **ftp:**[[*//username*[**:***password*]**@***location*]*/directory*]*/tar-filename***.tar**
- For the RCP, the syntax is **rcp:**[[*//username***@***location*]*/directory*]*/tar-filename***.tar**
- For the TFTP, the syntax is **tftp:**[[*//location*]*/directory*]*/tar-filename***.tar**

The *tar-filename***.tar** is the tar file to be created.

For **flash:***/file-url*, specify the location on the local flash file system from which the new tar file is created. You can also specify an optional list of files or directories within the source directory to write to the new tar file. If none are specified, all files and directories at this level are written to the newly created tar file.

This example shows how to create a tar file. This command writes the contents of the *new-configs* directory on the local flash device to a file named *saved.tar* on the TFTP server at 172.20.10.30:

```
Switch# archive tar /create tftp:172.20.10.30/saved.tar flash:/new-configs
```

Displaying the Contents of a tar File

To display the contents of a tar file on the screen, use this privileged EXEC command:

archive tar /table *source-url*

For *source-url*, specify the source URL alias for the local or network file system. These options are supported:

- For the local flash file system, the syntax is **flash:**
- For the FTP, the syntax is **ftp:**[[*//username*[**:***password*]**@***location*]*/directory*]*/tar-filename***.tar**
- For the RCP, the syntax is **rcp:**[[*//username***@***location*]*/directory*]*/tar-filename***.tar**
- For the TFTP, the syntax is **tftp:**[[*//location*]*/directory*]*/tar-filename***.tar**

The *tar-filename***.tar** is the tar file to display.

This example shows how to display the contents of a switch tar file that is in flash memory:

```
Switch# archive tar /table flash:me340x-metrobase-tar.122-25.EX.tar
info (219 bytes)
me340x-metrobase-mz.122-25.EX/ (directory)
me340x-metrobase-mz.122-25.EX/html/ (directory)
me340x-metrobase-mz.122-25.EX/html/foo.html (0 bytes)
me340x-metrobase-mz.122-25.EX/me340x-metrobase-mz.122-25.EX.bin (4527884 bytes)
me340x-metrobase-mz.122-25.EX/info (346 bytes)
info (110 bytes)
```

Extracting a tar File

To extract a tar file into a directory on the flash file system, use this privileged EXEC command:

archive tar /xtract *source-url* **flash:**/*file-url* [*dir/file...*]

For *source-url*, specify the source URL alias for the local file system. These options are supported:

- For the local flash file system, the syntax is **flash:**
- For the FTP, the syntax is **ftp:**[[//*username*[**:***password*]**@***location*]/*directory*]/*tar-filename***.tar**
- For the RCP, the syntax is **rcp:**[[//*username***@***location*]/*directory*]/*tar-filename***.tar**
- For the TFTP, the syntax is **tftp:**[[//*location*]/*directory*]/*tar-filename***.tar**

The *tar-filename***.tar** is the tar file from which to extract files.

For **flash:**/*file-url* [*dir/file...*], specify the location on the local flash file system into which the tar file is extracted. Use the *dir/file...* option to specify an optional list of files or directories within the tar file to be extracted. If none are specified, all files and directories are extracted.

This example shows how to extract the contents of a tar file located on the TFTP server at 172.20.10.30. This command extracts just the *new-configs* directory into the root directory on the local flash file system. The remaining files in the *saved.tar* file are ignored.

```
Switch# archive tar /xtract tftp:/172.20.10.30/saved.tar flash:/new-configs
```

Displaying the Contents of a File

To display the contents of any readable file, including a file on a remote file system, use the **more** [**/ascii** | **/binary** | **/ebcdic**] *file-url* privileged EXEC command:.

This example shows how to display the contents of a configuration file on a TFTP server:

```
Switch# more tftp://serverA/hampton/savedconfig
!
! Saved configuration on server
!
version 11.3
service timestamps log datetime localtime
service linenumber
service udp-small-servers
service pt-vty-logging
!
<output truncated>
```

Working with Configuration Files

This section describes how to create, load, and maintain configuration files.

Configuration files contain commands entered to customize the function of the Cisco IOS software. A way to create a basic configuration file is to use the **setup** program or to enter the **setup** privileged EXEC command. For more information, see Chapter 3, "Assigning the Switch IP Address and Default Gateway."

You can copy (*download*) configuration files from a TFTP, FTP, or RCP server to the running configuration or startup configuration of the switch. You might want to perform this for one of these reasons:

- To restore a backed-up configuration file.
- To use the configuration file for another switch. For example, you might add another switch to your network and want it to have a configuration similar to the original switch. By copying the file to the new switch, you can change the relevant parts rather than recreating the whole file.
- To load the same configuration commands on all the switches in your network so that all the switches have similar configurations.

You can copy (*upload*) configuration files from the switch to a file server by using TFTP, FTP, or RCP. You might perform this task to back up a current configuration file to a server before changing its contents so that you can later restore the original configuration file from the server.

The protocol you use depends on which type of server you are using. The FTP and RCP transport mechanisms provide faster performance and more reliable delivery of data than TFTP. These improvements are possible because FTP and RCP are built on and use the TCP/IP stack, which is connection-oriented.

These sections contain this configuration information:

- Guidelines for Creating and Using Configuration Files, page B-8
- Configuration File Types and Location, page B-9
- Creating a Configuration File By Using a Text Editor, page B-9
- Copying Configuration Files By Using TFTP, page B-10
- Copying Configuration Files By Using FTP, page B-11
- Copying Configuration Files By Using RCP, page B-15
- Clearing Configuration Information, page B-18

Guidelines for Creating and Using Configuration Files

Creating configuration files can aid in your switch configuration. Configuration files can contain some or all of the commands needed to configure one or more switches. For example, you might want to download the same configuration file to several switches that have the same hardware configuration.

Use these guidelines when creating a configuration file:

- We recommend that you connect through the console port for the initial configuration of the switch. If you are accessing the switch through a network connection instead of through a direct connection to the console port, keep in mind that some configuration changes (such as changing the switch IP address or disabling ports) can cause a loss of connectivity to the switch.
- If no password has been set on the switch, we recommend that you set one by using the **enable secret** *secret-password* global configuration command.

Note The **copy** {**ftp:** | **rcp:** | **tftp:**} **system:running-config** privileged EXEC command loads the configuration files on the switch as if you were entering the commands at the command line. The switch does not erase the existing running configuration before adding the commands. If a command in the copied configuration file replaces a command in the existing configuration file, the existing command is erased. For example, if the copied configuration file contains a different IP address in a particular command than the existing configuration, the IP address in the copied configuration is used. However, some commands in the existing configuration might not be replaced or negated. In this case, the resulting configuration file is a mixture of the existing configuration file and the copied configuration file, with the copied configuration file having precedence.

To restore a configuration file to an exact copy of a file stored on a server, copy the configuration file directly to the startup configuration (by using the **copy** {**ftp:** | **rcp:** | **tftp:**} **nvram:startup-config** privileged EXEC command), and reload the switch.

Configuration File Types and Location

Startup configuration files are used during system startup to configure the software. Running configuration files contain the current configuration of the software. The two configuration files can be different. For example, you might want to change the configuration for a short time period rather than permanently. In this case, you would change the running configuration but not save the configuration by using the **copy running-config startup-config** privileged EXEC command.

The running configuration is saved in DRAM; the startup configuration is stored in the NVRAM section of flash memory.

Creating a Configuration File By Using a Text Editor

When creating a configuration file, you must list commands logically so that the system can respond appropriately. This is one method of creating a configuration file:

Step 1 Copy an existing configuration from a switch to a server.

For more information, see the "Downloading the Configuration File By Using TFTP" section on page B-10, the "Downloading a Configuration File By Using FTP" section on page B-13, or the "Downloading a Configuration File By Using RCP" section on page B-16.

Step 2 Open the configuration file in a text editor, such as vi or emacs on UNIX or Notepad on a PC.

Step 3 Extract the portion of the configuration file with the desired commands, and save it in a new file.

Step 4 Copy the configuration file to the appropriate server location. For example, copy the file to the TFTP directory on the workstation (usually /tftpboot on a UNIX workstation).

Step 5 Make sure the permissions on the file are set to world-read.

Copying Configuration Files By Using TFTP

You can configure the switch by using configuration files you create, download from another switch, or download from a TFTP server. You can copy (upload) configuration files to a TFTP server for storage.

These sections contain this configuration information:

- Preparing to Download or Upload a Configuration File By Using TFTP, page B-10
- Downloading the Configuration File By Using TFTP, page B-10
- Uploading the Configuration File By Using TFTP, page B-11

Preparing to Download or Upload a Configuration File By Using TFTP

Before you begin downloading or uploading a configuration file by using TFTP, do these tasks:

- Ensure that the workstation acting as the TFTP server is properly configured. On a Sun workstation, make sure that the /etc/inetd.conf file contains this line:

  ```
  tftp dgram udp wait root /usr/etc/in.tftpd in.tftpd -p -s /tftpboot
  ```

 Make sure that the /etc/services file contains this line:

  ```
  tftp 69/udp
  ```

 Note You must restart the inetd daemon after modifying the /etc/inetd.conf and /etc/services files. To restart the daemon, either stop the inetd process and restart it, or enter a **fastboot** command (on the SunOS 4.x) or a **reboot** command (on Solaris 2.x or SunOS 5.x). For more information on the TFTP daemon, see the documentation for your workstation.

- Ensure that the switch has a route to the TFTP server. The switch and the TFTP server must be in the same subnetwork if you do not have a router to route traffic between subnets. Check connectivity to the TFTP server by using the **ping** command.
- Ensure that the configuration file to be downloaded is in the correct directory on the TFTP server (usually /tftpboot on a UNIX workstation).
- For download operations, ensure that the permissions on the file are set correctly. The permission on the file should be world-read.
- Before uploading the configuration file, you might need to create an empty file on the TFTP server. To create an empty file, enter the **touch** *filename* command, where *filename* is the name of the file you will use when uploading it to the server.
- During upload operations, if you are overwriting an existing file (including an empty file, if you had to create one) on the server, ensure that the permissions on the file are set correctly. Permissions on the file should be world-write.

Downloading the Configuration File By Using TFTP

To configure the switch by using a configuration file downloaded from a TFTP server, follow these steps:

Step 1 Copy the configuration file to the appropriate TFTP directory on the workstation.

Step 2 Verify that the TFTP server is properly configured by referring to the "Preparing to Download or Upload a Configuration File By Using TFTP" section on page B-10.

Step 3 Log into the switch through the console port or a Telnet session.

Step 4 Download the configuration file from the TFTP server to configure the switch.

Specify the IP address or hostname of the TFTP server and the name of the file to download.

Use one of these privileged EXEC commands:

- **copy tftp:**[[[*//location*]*/directory*]*/filename*] **system:running-config**
- **copy tftp:**[[[*//location*]*/directory*]*/filename*] **nvram:startup-config**

The configuration file downloads, and the commands are executed as the file is parsed line-by-line.

This example shows how to configure the software from the file *tokyo-confg* at IP address 172.16.2.155:

```
Switch# copy tftp://172.16.2.155/tokyo-confg system:running-config
Configure using tokyo-confg from 172.16.2.155? [confirm] y
Booting tokyo-confg from 172.16.2.155:!!! [OK - 874/16000 bytes]
```

Uploading the Configuration File By Using TFTP

To upload a configuration file from a switch to a TFTP server for storage, follow these steps:

Step 1 Verify that the TFTP server is properly configured by referring to the "Preparing to Download or Upload a Configuration File By Using TFTP" section on page B-10.

Step 2 Log into the switch through the console port or a Telnet session.

Step 3 Upload the switch configuration to the TFTP server. Specify the IP address or hostname of the TFTP server and the destination filename.

Use one of these privileged EXEC commands:

- **copy system:running-config tftp:**[[[*//location*]*/directory*]*/filename*]
- **copy nvram:startup-config tftp:**[[[*//location*]*/directory*]*/filename*]

The file is uploaded to the TFTP server.

This example shows how to upload a configuration file from a switch to a TFTP server:

```
Switch# copy system:running-config tftp://172.16.2.155/tokyo-confg
Write file tokyo-confg on host 172.16.2.155? [confirm] y
#
Writing tokyo-confg!!! [OK]
```

Copying Configuration Files By Using FTP

You can copy configuration files to or from an FTP server.

The FTP protocol requires a client to send a remote username and password on each FTP request to a server.

When you copy a configuration file from the switch to a server by using FTP, the Cisco IOS software sends the first valid username in this list:

- The username specified in the **copy** command if a username is specified.
- The username set by the **ip ftp username** *username* global configuration command if the command is configured.
- Anonymous.

The switch sends the first valid password in this list:

- The password specified in the **copy** command if a password is specified.
- The password set by the **ip ftp password** *password* global configuration command if the command is configured.
- The switch forms a password named *username@switchname.domain*. The variable *username* is the username associated with the current session, *switchname* is the configured hostname, and *domain* is the domain of the switch.

The username and password must be associated with an account on the FTP server. If you are writing to the server, the FTP server must be properly configured to accept your FTP write request.

Use the **ip ftp username** and **ip ftp password** commands to specify a username and password for all copies. Include the username in the **copy** command if you want to specify only a username for that copy operation.

If the server has a directory structure, the configuration file is written to or copied from the directory associated with the username on the server. For example, if the configuration file resides in the home directory of a user on the server, specify that user's name as the remote username.

For more information, see the documentation for your FTP server.

These sections contain this configuration information:

- Preparing to Download or Upload a Configuration File By Using FTP, page B-12
- Downloading a Configuration File By Using FTP, page B-13
- Uploading a Configuration File By Using FTP, page B-14

Preparing to Download or Upload a Configuration File By Using FTP

Before you begin downloading or uploading a configuration file by using FTP, do these tasks:

- Ensure that the switch has a route to the FTP server. The switch and the FTP server must be in the same subnetwork if you do not have a router to route traffic between subnets. Check connectivity to the FTP server by using the **ping** command.
- If you are accessing the switch through the console or a Telnet session and you do not have a valid username, make sure that the current FTP username is the one that you want to use for the FTP download. You can enter the **show users** privileged EXEC command to view the valid username. If you do not want to use this username, create a new FTP username by using the **ip ftp username** *username* global configuration command during all copy operations. The new username is stored in NVRAM. If you are accessing the switch through a Telnet session and you have a valid username, this username is used, and you do not need to set the FTP username. Include the username in the **copy** command if you want to specify a username for only that copy operation.
- When you upload a configuration file to the FTP server, it must be properly configured to accept the write request from the user on the switch.

For more information, see the documentation for your FTP server.

Downloading a Configuration File By Using FTP

Beginning in privileged EXEC mode, follow these steps to download a configuration file by using FTP:

	Command	Purpose
Step 1		Verify that the FTP server is properly configured by referring to the "Preparing to Download or Upload a Configuration File By Using FTP" section on page B-12.
Step 2		Log into the switch through the console port or a Telnet session.
Step 3	**configure terminal**	Enter global configuration mode on the switch. This step is required only if you override the default remote username or password (see Steps 4, 5, and 6).
Step 4	**ip ftp username** *username*	(Optional) Change the default remote username.
Step 5	**ip ftp password** *password*	(Optional) Change the default password.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**copy ftp:**[[[//[*username*[**:***password*]**@**]*location*]/*directory*] /*filename*] **system:running-config** or **copy ftp:**[[[//[*username*[**:***password*]**@**]*location*]/*directory*] /*filename*] **nvram:startup-config**	Using FTP, copy the configuration file from a network server to the running configuration or to the startup configuration file.

This example shows how to copy a configuration file named *host1-confg* from the *netadmin1* directory on the remote server with an IP address of 172.16.101.101 and to load and run those commands on the switch:

```
Switch# copy ftp://netadmin1:mypass@172.16.101.101/host1-confg system:running-config
Configure using host1-confg from 172.16.101.101? [confirm]
Connected to 172.16.101.101
Loading 1112 byte file host1-confg:![OK]
Switch#
%SYS-5-CONFIG: Configured from host1-config by ftp from 172.16.101.101
```

This example shows how to specify a remote username of *netadmin1*. The software copies the configuration file *host2-confg* from the *netadmin1* directory on the remote server with an IP address of 172.16.101.101 to the switch startup configuration.

```
Switch# configure terminal
Switch(config)# ip ftp username netadmin1
Switch(config)# ip ftp password mypass
Switch(config)# end
Switch# copy ftp: nvram:startup-config
Address of remote host [255.255.255.255]? 172.16.101.101
Name of configuration file[rtr2-confg]? host2-confg
Configure using host2-confg from 172.16.101.101?[confirm]
Connected to 172.16.101.101
Loading 1112 byte file host2-confg:![OK]
[OK]
Switch#
%SYS-5-CONFIG_NV:Non-volatile store configured from host2-config by ftp from
172.16.101.101
```

Uploading a Configuration File By Using FTP

Beginning in privileged EXEC mode, follow these steps to upload a configuration file by using FTP:

	Command	Purpose
Step 1		Verify that the FTP server is properly configured by referring to the "Preparing to Download or Upload a Configuration File By Using FTP" section on page B-12.
Step 2		Log into the switch through the console port or a Telnet session.
Step 3	**configure terminal**	Enter global configuration mode. This step is required only if you override the default remote username or password (see Steps 4, 5, and 6).
Step 4	**ip ftp username** *username*	(Optional) Change the default remote username.
Step 5	**ip ftp password** *password*	(Optional) Change the default password.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**copy system:running-config ftp:**[[[//[*username*[**:***password*]**@**]*location*]/*directory*] /*filename*] or **copy nvram:startup-config ftp:**[[[//[*username*[**:***password*]**@**]*location*]/*directory*] /*filename*]	Using FTP, store the switch running or startup configuration file to the specified location.

This example shows how to copy the running configuration file named *switch2-confg* to the *netadmin1* directory on the remote host with an IP address of 172.16.101.101:

```
Switch# copy system:running-config ftp://netadmin1:mypass@172.16.101.101/switch2-confg
Write file switch2-confg on host 172.16.101.101?[confirm]
Building configuration...[OK]
Connected to 172.16.101.101
Switch#
```

This example shows how to store a startup configuration file on a server by using FTP to copy the file:

```
Switch# configure terminal
Switch(config)# ip ftp username netadmin2
Switch(config)# ip ftp password mypass
Switch(config)# end
Switch# copy nvram:startup-config ftp:
Remote host[]? 172.16.101.101
Name of configuration file to write [switch2-confg]?
Write file switch2-confg on host 172.16.101.101?[confirm]
![OK]
```

Copying Configuration Files By Using RCP

The RCP provides another method of downloading, uploading, and copying configuration files between remote hosts and the switch. Unlike TFTP, which uses User Datagram Protocol (UDP), a connectionless protocol, RCP uses TCP, which is connection-oriented.

To use RCP to copy files, the server from or to which you will be copying files must support RCP. The RCP copy commands rely on the rsh server (or daemon) on the remote system. To copy files by using RCP, you do not need to create a server for file distribution as you do with TFTP. You only need to have access to a server that supports the remote shell (rsh). (Most UNIX systems support rsh.) Because you are copying a file from one place to another, you must have read permission on the source file and write permission on the destination file. If the destination file does not exist, RCP creates it for you.

The RCP requires a client to send a remote username with each RCP request to a server. When you copy a configuration file from the switch to a server, the Cisco IOS software sends the first valid username in this list:

- The username specified in the **copy** command if a username is specified.
- The username set by the **ip rcmd remote-username** *username* global configuration command if the command is configured.
- The remote username associated with the current TTY (terminal) process. For example, if the user is connected to the router through Telnet and was authenticated through the **username** command, the switch software sends the Telnet username as the remote username.
- The switch hostname.

For a successful RCP copy request, you must define an account on the network server for the remote username. If the server has a directory structure, the configuration file is written to or copied from the directory associated with the remote username on the server. For example, if the configuration file is in the home directory of a user on the server, specify that user's name as the remote username.

These sections contain this configuration information:

- Preparing to Download or Upload a Configuration File By Using RCP, page B-15
- Downloading a Configuration File By Using RCP, page B-16
- Uploading a Configuration File By Using RCP, page B-17

Preparing to Download or Upload a Configuration File By Using RCP

Before you begin downloading or uploading a configuration file by using RCP, do these tasks:

- Ensure that the workstation acting as the RCP server supports the remote shell (rsh).
- Ensure that the switch has a route to the RCP server. The switch and the server must be in the same subnetwork if you do not have a router to route traffic between subnets. Check connectivity to the RCP server by using the **ping** command.
- If you are accessing the switch through the console or a Telnet session and you do not have a valid username, make sure that the current RCP username is the one that you want to use for the RCP download. You can enter the **show users** privileged EXEC command to view the valid username. If you do not want to use this username, create a new RCP username by using the **ip rcmd remote-username** *username* global configuration command to be used during all copy operations. The new username is stored in NVRAM. If you are accessing the switch through a Telnet session and you have a valid username, this username is used, and you do not need to set the RCP username. Include the username in the **copy** command if you want to specify a username for only that copy operation.

- When you upload a file to the RCP server, it must be properly configured to accept the RCP write request from the user on the switch. For UNIX systems, you must add an entry to the .rhosts file for the remote user on the RCP server. For example, suppose that the switch contains these configuration lines:

  ```
  hostname Switch1
  ip rcmd remote-username User0
  ```

 If the switch IP address translates to *Switch1.company.com*, the .rhosts file for User0 on the RCP server should contain this line:

  ```
  Switch1.company.com Switch1
  ```

For more information, see the documentation for your RCP server.

Downloading a Configuration File By Using RCP

Beginning in privileged EXEC mode, follow these steps to download a configuration file by using RCP:

	Command	Purpose
Step 1		Verify that the RCP server is properly configured by referring to the "Preparing to Download or Upload a Configuration File By Using RCP" section on page B-15.
Step 2		Log into the switch through the console port or a Telnet session.
Step 3	**configure terminal**	Enter global configuration mode. This step is required only if you override the default remote username (see Steps 4 and 5).
Step 4	**ip rcmd remote-username** *username*	(Optional) Specify the remote username.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**copy rcp:**[[[//[*username***@**]*location*]/*directory*]/*filename*] **system:running-config** or **copy rcp:**[[[//[*username***@**]*location*]/*directory*]/*filename*] **nvram:startup-config**	Using RCP, copy the configuration file from a network server to the running configuration or to the startup configuration file.

This example shows how to copy a configuration file named *host1-confg* from the *netadmin1* directory on the remote server with an IP address of 172.16.101.101 and load and run those commands on the switch:

```
Switch# copy rcp://netadmin1@172.16.101.101/host1-confg system:running-config
Configure using host1-confg from 172.16.101.101? [confirm]
Connected to 172.16.101.101
Loading 1112 byte file host1-confg:![OK]
Switch#
%SYS-5-CONFIG: Configured from host1-config by rcp from 172.16.101.101
```

This example shows how to specify a remote username of *netadmin1*. Then it copies the configuration file *host2-confg* from the *netadmin1* directory on the remote server with an IP address of 172.16.101.101 to the startup configuration:

```
Switch# configure terminal
Switch(config)# ip rcmd remote-username netadmin1
Switch(config)# end
Switch# copy rcp: nvram:startup-config
Address of remote host [255.255.255.255]? 172.16.101.101
Name of configuration file[rtr2-confg]? host2-confg
Configure using host2-confg from 172.16.101.101?[confirm]
Connected to 172.16.101.101
Loading 1112 byte file host2-confg:![OK]
[OK]
Switch#
%SYS-5-CONFIG_NV:Non-volatile store configured from host2-config by rcp from
172.16.101.101
```

Uploading a Configuration File By Using RCP

Beginning in privileged EXEC mode, follow these steps to upload a configuration file by using RCP:

	Command	Purpose
Step 1		Verify that the RCP server is properly configured by referring to the "Preparing to Download or Upload a Configuration File By Using RCP" section on page B-15.
Step 2		Log into the switch through the console port or a Telnet session.
Step 3	**configure terminal**	Enter global configuration mode. This step is required only if you override the default remote username (see Steps 4 and 5).
Step 4	**ip rcmd remote-username** *username*	(Optional) Specify the remote username.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**copy system:running-config rcp:**[[[//[*username*@]*location*]/*directory*]/*filename*] or **copy nvram:startup-config rcp:**[[[//[*username*@]*location*]/*directory*]/*filename*]	Using RCP, copy the configuration file from a switch running or startup configuration file to a network server.

This example shows how to copy the running configuration file named *switch2-confg* to the *netadmin1* directory on the remote host with an IP address of 172.16.101.101:

```
Switch# copy system:running-config rcp://netadmin1@172.16.101.101/switch2-confg
Write file switch-confg on host 172.16.101.101?[confirm]
Building configuration...[OK]
Connected to 172.16.101.101
Switch#
```

This example shows how to store a startup configuration file on a server:

```
Switch# configure terminal
Switch(config)# ip rcmd remote-username netadmin2
Switch(config)# end
Switch# copy nvram:startup-config rcp:
Remote host[]? 172.16.101.101
Name of configuration file to write [switch2-confg]?
Write file switch2-confg on host 172.16.101.101?[confirm]
![OK]
```

Clearing Configuration Information

You can clear the configuration information from the startup configuration. If you reboot the switch with no startup configuration, the switch enters the setup program so that you can reconfigure the switch with all new settings.

Clearing the Startup Configuration File

To clear the contents of your startup configuration, use the **erase nvram:** or the **erase startup-config** privileged EXEC command.

Caution You cannot restore the startup configuration file after it has been deleted.

Deleting a Stored Configuration File

To delete a saved configuration from flash memory, use the **delete flash:***filename* privileged EXEC command. Depending on the setting of the **file prompt** global configuration command, you might be prompted for confirmation before you delete a file. By default, the switch prompts for confirmation on destructive file operations. For more information about the **file prompt** command, see the *Cisco IOS Command Reference for Release 12.2*.

Caution You cannot restore a file after it has been deleted.

Working with Software Images

This section describes how to archive (download and upload) software image files, which contain the system software, the Cisco IOS code, and the embedded device manager software.

Note Instead of using the **copy** privileged EXEC command or the **archive tar** privileged EXEC command, we recommend using the **archive download-sw** and **archive upload-sw** privileged EXEC commands to download and upload software image files.

You can download a switch image file from a TFTP, FTP, or RCP server to upgrade the switch software. For information about upgrading your switch by using a TFTP server, see the release notes

You can replace the current image with the new one or keep the current image in flash memory after a download.

You upload a switch image file to a TFTP, FTP, or RCP server for backup purposes. You can use this uploaded image for future downloads to the same switch or to another of the same type.

The protocol that you use depends on which type of server you are using. The FTP and RCP transport mechanisms provide faster performance and more reliable delivery of data than TFTP. These improvements are possible because FTP and RCP are built on and use the TCP/IP stack, which is connection-oriented.

These sections contain this configuration information:

- Image Location on the Switch, page B-19
- tar File Format of Images on a Server or Cisco.com, page B-19
- Copying Image Files By Using TFTP, page B-20
- Copying Image Files By Using FTP, page B-23
- Copying Image Files By Using RCP, page B-28

Note For a list of software images and the supported upgrade paths, see the release notes for your switch.

Image Location on the Switch

The Cisco IOS image is stored as a *.bin* file in a directory that shows the version number. A subdirectory contains the files needed for web management. The image is stored on the system board flash memory (flash:).

You can use the **show version** privileged EXEC command to see the software version that is currently running on your switch. In the display, check the line that begins with `System image file is...`. It shows the directory name in flash memory where the image is stored.

You can also use the **dir** *filesystem***:** privileged EXEC command to see the directory names of other software images that you might have stored in flash memory.

tar File Format of Images on a Server or Cisco.com

Software images located on a server or downloaded from Cisco.com are provided in a tar file format, which contains these files:

- An *info* file, which serves as a table of contents for the tar file
- One or more subdirectories containing other images and files, such as Cisco IOS images

This example shows some of the information contained in the info file. Table B-3 provides additional details about this information:

```
version_suffix: metroipaccess-122-0.68.EX
version_directory: me340x-metroipaccess-mz.122-0.68.EX
image_system_type_id: 0x00000000
image_name: me340x-metroipaccess-mz.122-0.68.EX.bin
ios_image_file_size: 6277632
total_image_file_size: 6277632
image_feature: LAYER_2|MIN_DRAM_MEG=128
image_family: ME340x
stacking_number: 1.11
board_ids: 0x00000028
info_end:
```

Note Disregard the stacking_number field. It does not apply to the switch.

Table B-3 info File Description

Field	Description
version_suffix	Specifies the Cisco IOS image version string suffix
version_directory	Specifies the directory where the Cisco IOS image and the HTML subdirectory are installed
image_name	Specifies the name of the Cisco IOS image within the tar file
ios_image_file_size	Specifies the Cisco IOS image size in the tar file, which is an approximate measure of how much flash memory is required to hold just the Cisco IOS image
total_image_file_size	Specifies the size of all the images (the Cisco IOS image and the web management files) in the tar file, which is an approximate measure of how much flash memory is required to hold them
image_feature	Describes the core functionality of the image
image_min_dram	Specifies the minimum amount of DRAM needed to run this image
image_family	Describes the family of products on which the software can be installed

Copying Image Files By Using TFTP

You can download a switch image from a TFTP server or upload the image from the switch to a TFTP server.

You download a switch image file from a server to upgrade the switch software. You can overwrite the current image with the new one or keep the current image after a download.

You upload a switch image file to a server for backup purposes; this uploaded image can be used for future downloads to the same or another switch of the same type.

Note Instead of using the **copy** privileged EXEC command or the **archive tar** privileged EXEC command, we recommend using the **archive download-sw** and **archive upload-sw** privileged EXEC commands to download and upload software image files.

These sections contain this configuration information:

- Preparing to Download or Upload an Image File By Using TFTP, page B-21
- Downloading an Image File By Using TFTP, page B-21
- Uploading an Image File By Using TFTP, page B-23

Preparing to Download or Upload an Image File By Using TFTP

Before you begin downloading or uploading an image file by using TFTP, do these tasks:

- Ensure that the workstation acting as the TFTP server is properly configured. On a Sun workstation, make sure that the /etc/inetd.conf file contains this line:

  ```
  tftp dgram udp wait root /usr/etc/in.tftpd in.tftpd -p -s /tftpboot
  ```

 Make sure that the /etc/services file contains this line:

  ```
  tftp 69/udp
  ```

 Note You must restart the inetd daemon after modifying the /etc/inetd.conf and /etc/services files. To restart the daemon, either stop the inetd process and restart it, or enter a **fastboot** command (on the SunOS 4.x) or a **reboot** command (on Solaris 2.x or SunOS 5.x). For more information on the TFTP daemon, see the documentation for your workstation.

- Ensure that the switch has a route to the TFTP server. The switch and the TFTP server must be in the same subnetwork if you do not have a router to route traffic between subnets. Check connectivity to the TFTP server by using the **ping** command.
- Ensure that the image to be downloaded is in the correct directory on the TFTP server (usually /tftpboot on a UNIX workstation).
- For download operations, ensure that the permissions on the file are set correctly. The permission on the file should be world-read.
- Before uploading the image file, you might need to create an empty file on the TFTP server. To create an empty file, enter the **touch** *filename* command, where *filename* is the name of the file you will use when uploading the image to the server.
- During upload operations, if you are overwriting an existing file (including an empty file, if you had to create one) on the server, ensure that the permissions on the file are set correctly. Permissions on the file should be world-write.

Downloading an Image File By Using TFTP

You can download a new image file and replace the current image or keep the current image.

Beginning in privileged EXEC mode, follow Steps 1 through 3 to download a new image from a TFTP server and overwrite the existing image. To keep the current image, go to Step 3.

	Command	Purpose
Step 1		Copy the image to the appropriate TFTP directory on the workstation. Make sure the TFTP server is properly configured; see the "Preparing to Download or Upload an Image File By Using TFTP" section on page B-21.
Step 2		Log into the switch through the console port or a Telnet session.

	Command	Purpose
Step 3	**archive download-sw /overwrite /reload tftp:**[[//*location*]/*directory*]/*image-name*.**tar**	Download the image file from the TFTP server to the switch, and overwrite the current image. • The **/overwrite** option overwrites the software image in flash memory with the downloaded image. • The **/reload** option reloads the system after downloading the image unless the configuration has been changed and not been saved. • For //*location*, specify the IP address of the TFTP server. • For /*directory*/*image-name*.**tar**, specify the directory (optional) and the image to download. Directory and image names are case sensitive.
Step 4	**archive download-sw /leave-old-sw /reload tftp:**[[//*location*]/*directory*]/*image-name*.**tar**	Download the image file from the TFTP server to the switch, and keep the current image. • The **/leave-old-sw** option keeps the old software version after a download. • The **/reload** option reloads the system after downloading the image unless the configuration has been changed and not been saved. • For //*location*, specify the IP address of the TFTP server. • For /*directory*/*image-name*.**tar**, specify the directory (optional) and the image to download. Directory and image names are case sensitive.

The download algorithm verifies that the image is appropriate for the switch model and that enough DRAM is present, or it aborts the process and reports an error. If you specify the **/overwrite** option, the download algorithm removes the existing image on the flash device whether or not it is the same as the new one, downloads the new image, and then reloads the software.

Note If the flash device has sufficient space to hold two images and you want to overwrite one of these images with the same version, you must specify the **/overwrite** option.

If you specify the **/leave-old-sw**, the existing files are not removed. If there is not enough space to install the new image and keep the current running image, the download process stops, and an error message is displayed.

The algorithm installs the downloaded image on the system board flash device (flash:). The image is placed into a new directory named with the software version string, and the BOOT environment variable is updated to point to the newly installed image.

If you kept the old image during the download process (you specified the **/leave-old-sw** keyword), you can remove it by entering the **delete /force /recursive** *filesystem***:/***file-url* privileged EXEC command. For *filesystem*, use **flash:** for the system board flash device. For *file-url*, enter the directory name of the old image. All the files in the directory and the directory are removed.

Caution For the download and upload algorithms to operate properly, do *not* rename image names.

Uploading an Image File By Using TFTP

You can upload an image from the switch to a TFTP server. You can later download this image to the switch or to another switch of the same type.

Use the upload feature only if the web management pages associated with the embedded device manager have been installed with the existing image.

Beginning in privileged EXEC mode, follow these steps to upload an image to a TFTP server:

	Command	Purpose
Step 1		Make sure the TFTP server is properly configured; see the "Preparing to Download or Upload an Image File By Using TFTP" section on page B-21.
Step 2		Log into the switch through the console port or a Telnet session.
Step 3	**archive upload-sw tftp:**[[*//location*]*/directory*]*/image-name***.tar**	Upload the currently running switch image to the TFTP server. • For *//location*, specify the IP address of the TFTP server. • For */directory/image-name***.tar**, specify the directory (optional) and the name of the software image to be uploaded. Directory and image names are case sensitive. The *image-name***.tar** is the name of the software image to be stored on the server.

The **archive upload-sw** privileged EXEC command builds an image file on the server by uploading these files in order: info, the Cisco IOS image, and the web management files. After these files are uploaded, the upload algorithm creates the tar file format.

Caution For the download and upload algorithms to operate properly, do *not* rename image names.

Copying Image Files By Using FTP

You can download a switch image from an FTP server or upload the image from the switch to an FTP server.

You download a switch image file from a server to upgrade the switch software. You can overwrite the current image with the new one or keep the current image after a download.

You upload a switch image file to a server for backup purposes. You can use this uploaded image for future downloads to the switch or another switch of the same type.

Note Instead of using the **copy** privileged EXEC command or the **archive tar** privileged EXEC command, we recommend using the **archive download-sw** and **archive upload-sw** privileged EXEC commands to download and upload software image files.

These sections contain this configuration information:

- Preparing to Download or Upload an Image File By Using FTP, page B-24
- Downloading an Image File By Using FTP, page B-25
- Uploading an Image File By Using FTP, page B-27

Preparing to Download or Upload an Image File By Using FTP

You can copy images files to or from an FTP server.

The FTP protocol requires a client to send a remote username and password on each FTP request to a server. When you copy an image file from the switch to a server by using FTP, the Cisco IOS software sends the first valid username in this list:

- The username specified in the **archive download-sw** or **archive upload-sw** privileged EXEC command if a username is specified.
- The username set by the **ip ftp username** *username* global configuration command if the command is configured.
- Anonymous.

The switch sends the first valid password in this list:

- The password specified in the **archive download-sw** or **archive upload-sw** privileged EXEC command if a password is specified.
- The password set by the **ip ftp password** *password* global configuration command if the command is configured.
- The switch forms a password named *username@switchname.domain*. The variable *username* is the username associated with the current session, *switchname* is the configured hostname, and *domain* is the domain of the switch.

The username and password must be associated with an account on the FTP server. If you are writing to the server, the FTP server must be properly configured to accept the FTP write request from you.

Use the **ip ftp username** and **ip ftp password** commands to specify a username and password for all copies. Include the username in the **archive download-sw** or **archive upload-sw** privileged EXEC command if you want to specify a username only for that operation.

If the server has a directory structure, the image file is written to or copied from the directory associated with the username on the server. For example, if the image file resides in the home directory of a user on the server, specify that user's name as the remote username.

Before you begin downloading or uploading an image file by using FTP, do these tasks:

- Ensure that the switch has a route to the FTP server. The switch and the FTP server must be in the same subnetwork if you do not have a router to route traffic between subnets. Check connectivity to the FTP server by using the **ping** command.
- If you are accessing the switch through the console or a Telnet session and you do not have a valid username, make sure that the current FTP username is the one that you want to use for the FTP download. You can enter the **show users** privileged EXEC command to view the valid username. If you do not want to use this username, create a new FTP username by using the **ip ftp username** *username* global configuration command. This new name will be used during all archive operations. The new username is stored in NVRAM. If you are accessing the switch through a Telnet session and you have a valid username, this username is used, and you do not need to set the FTP username. Include the username in the **archive download-sw** or **archive upload-sw** privileged EXEC command if you want to specify a username for that operation only.
- When you upload an image file to the FTP server, it must be properly configured to accept the write request from the user on the switch.

For more information, see the documentation for your FTP server.

Downloading an Image File By Using FTP

You can download a new image file and overwrite the current image or keep the current image.

Beginning in privileged EXEC mode, follow Steps 1 through 7 to download a new image from an FTP server and overwrite the existing image. To keep the current image, go to Step 7.

	Command	Purpose
Step 1		Verify that the FTP server is properly configured by referring to the "Preparing to Download or Upload an Image File By Using FTP" section on page B-24.
Step 2		Log into the switch through the console port or a Telnet session.
Step 3	**configure terminal**	Enter global configuration mode. This step is required only if you override the default remote username or password (see Steps 4, 5, and 6).
Step 4	**ip ftp username** *username*	(Optional) Change the default remote username.
Step 5	**ip ftp password** *password*	(Optional) Change the default password.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**archive download-sw /overwrite /reload ftp:**[[*//username*[**:***password*]**@***location*]*/directory*] */image-name***.tar**	Download the image file from the FTP server to the switch, and overwrite the current image. • The **/overwrite** option overwrites the software image in flash memory with the downloaded image. • The **/reload** option reloads the system after downloading the image unless the configuration has been changed and not been saved. • For *//username*[**:***password*], specify the username and password; these must be associated with an account on the FTP server. For more information, see the "Preparing to Download or Upload an Image File By Using FTP" section on page B-24. • For **@***location*, specify the IP address of the FTP server. • For *directory*/*image-name***.tar**, specify the directory (optional) and the image to download. Directory and image names are case sensitive.

	Command	Purpose
Step 8	**archive download-sw /leave-old-sw /reload ftp:**[[*//username*[**:***password*]**@***location*]*/directory*] */image-name*.**tar**	Download the image file from the FTP server to the switch, and keep the current image. • The **/leave-old-sw** option keeps the old software version after a download. • The **/reload** option reloads the system after downloading the image unless the configuration has been changed and not been saved. • For *//username*[**:***password*], specify the username and password. These must be associated with an account on the FTP server. For more information, see the "Preparing to Download or Upload an Image File By Using FTP" section on page B-24. • For **@***location*, specify the IP address of the FTP server. • For *directory/image-name*.**tar**, specify the directory (optional) and the image to download. Directory and image names are case sensitive.

The download algorithm verifies that the image is appropriate for the switch model and that enough DRAM is present, or it aborts the process and reports an error. If you specify the **/overwrite** option, the download algorithm removes the existing image on the flash device, whether or not it is the same as the new one, downloads the new image, and then reloads the software.

Note If the flash device has sufficient space to hold two images and you want to overwrite one of these images with the same version, you must specify the **/overwrite** option.

If you specify the **/leave-old-sw**, the existing files are not removed. If there is not enough space to install the new image and keep the running image, the download process stops, and an error message is displayed.

The algorithm installs the downloaded image onto the system board flash device (flash:). The image is placed into a new directory named with the software version string, and the BOOT environment variable is updated to point to the newly installed image.

If you kept the old image during the download process (you specified the **/leave-old-sw** keyword), you can remove it by entering the **delete /force /recursive** *filesystem***:/***file-url* privileged EXEC command. For *filesystem*, use **flash:** for the system board flash device. For *file-url*, enter the directory name of the old software image. All the files in the directory and the directory are removed.

Caution For the download and upload algorithms to operate properly, do *not* rename image names.

Uploading an Image File By Using FTP

You can upload an image from the switch to an FTP server. You can later download this image to the same switch or to another switch of the same type.

Use the upload feature only if the web management pages associated with the embedded device manager have been installed with the existing image.

Beginning in privileged EXEC mode, follow these steps to upload an image to an FTP server:

	Command	Purpose
Step 1		Verify that the FTP server is properly configured by referring to the "Preparing to Download or Upload a Configuration File By Using FTP" section on page B-12.
Step 2		Log into the switch through the console port or a Telnet session.
Step 3	**configure terminal**	Enter global configuration mode. This step is required only if you override the default remote username or password (see Steps 4, 5, and 6).
Step 4	**ip ftp username** *username*	(Optional) Change the default remote username.
Step 5	**ip ftp password** *password*	(Optional) Change the default password.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**archive upload-sw ftp:**[[//[*username*[**:***password*]**@**]*location*]/*directory*]/*image-name***.tar**	Upload the currently running switch image to the FTP server. • For *//username**:**password*, specify the username and password. These must be associated with an account on the FTP server. For more information, see the "Preparing to Download or Upload an Image File By Using FTP" section on page B-24. • For *@location*, specify the IP address of the FTP server. • For */directory/image-name***.tar**, specify the directory (optional) and the name of the software image to be uploaded. Directory and image names are case sensitive. The *image-name***.tar** is the name of the software image to be stored on the server.

The **archive upload-sw** command builds an image file on the server by uploading these files in order: info, the Cisco IOS image, and the web management files. After these files are uploaded, the upload algorithm creates the tar file format.

Caution For the download and upload algorithms to operate properly, do *not* rename image names.

Copying Image Files By Using RCP

You can download a switch image from an RCP server or upload the image from the switch to an RCP server.

You download a switch image file from a server to upgrade the switch software. You can overwrite the current image with the new one or keep the current image after a download.

You upload a switch image file to a server for backup purposes. You can use this uploaded image for future downloads to the same switch or another of the same type.

Note Instead of using the **copy** privileged EXEC command or the **archive tar** privileged EXEC command, we recommend using the **archive download-sw** and **archive upload-sw** privileged EXEC commands to download and upload software image files.

These sections contain this configuration information:

- Preparing to Download or Upload an Image File By Using RCP, page B-28
- Downloading an Image File By Using RCP, page B-29
- Uploading an Image File By Using RCP, page B-31

Preparing to Download or Upload an Image File By Using RCP

RCP provides another method of downloading and uploading image files between remote hosts and the switch. Unlike TFTP, which uses User Datagram Protocol (UDP), a connectionless protocol, RCP uses TCP, which is connection-oriented.

To use RCP to copy files, the server from or to which you will be copying files must support RCP. The RCP copy commands rely on the rsh server (or daemon) on the remote system. To copy files by using RCP, you do not need to create a server for file distribution as you do with TFTP. You only need to have access to a server that supports the remote shell (rsh). (Most UNIX systems support rsh.) Because you are copying a file from one place to another, you must have read permission on the source file and write permission on the destination file. If the destination file does not exist, RCP creates it for you.

RCP requires a client to send a remote username on each RCP request to a server. When you copy an image from the switch to a server by using RCP, the Cisco IOS software sends the first valid username in this list:

- The username specified in the **archive download-sw** or **archive upload-sw** privileged EXEC command if a username is specified.
- The username set by the **ip rcmd remote-username** *username* global configuration command if the command is entered.
- The remote username associated with the current TTY (terminal) process. For example, if the user is connected to the router through Telnet and was authenticated through the **username** command, the switch software sends the Telnet username as the remote username.
- The switch hostname.

For the RCP copy request to execute successfully, an account must be defined on the network server for the remote username. If the server has a directory structure, the image file is written to or copied from the directory associated with the remote username on the server. For example, if the image file resides in the home directory of a user on the server, specify that user's name as the remote username.

Before you begin downloading or uploading an image file by using RCP, do these tasks:

- Ensure that the workstation acting as the RCP server supports the remote shell (rsh).
- Ensure that the switch has a route to the RCP server. The switch and the server must be in the same subnetwork if you do not have a router to route traffic between subnets. Check connectivity to the RCP server by using the **ping** command.
- If you are accessing the switch through the console or a Telnet session and you do not have a valid username, make sure that the current RCP username is the one that you want to use for the RCP download. You can enter the **show users** privileged EXEC command to view the valid username. If you do not want to use this username, create a new RCP username by using the **ip rcmd remote-username** *username* global configuration command to be used during all archive operations. The new username is stored in NVRAM. If you are accessing the switch through a Telnet session and you have a valid username, this username is used, and there is no need to set the RCP username. Include the username in the **archive download-sw** or **archive upload-sw** privileged EXEC command if you want to specify a username only for that operation.
- When you upload an image to the RCP to the server, it must be properly configured to accept the RCP write request from the user on the switch. For UNIX systems, you must add an entry to the .rhosts file for the remote user on the RCP server. For example, suppose the switch contains these configuration lines:

  ```
  hostname Switch1
  ip rcmd remote-username User0
  ```

 If the switch IP address translates to *Switch1.company.com*, the .rhosts file for User0 on the RCP server should contain this line:

  ```
  Switch1.company.com Switch1
  ```

For more information, see the documentation for your RCP server.

Downloading an Image File By Using RCP

You can download a new image file and replace or keep the current image.

Beginning in privileged EXEC mode, follow Steps 1 through 6 to download a new image from an RCP server and overwrite the existing image. To keep the current image, go to Step 6.

	Command	Purpose
Step 1		Verify that the RCP server is properly configured by referring to the "Preparing to Download or Upload an Image File By Using RCP" section on page B-28.
Step 2		Log into the switch through the console port or a Telnet session.
Step 3	**configure terminal**	Enter global configuration mode. This step is required only if you override the default remote username (see Steps 4 and 5).
Step 4	**ip rcmd remote-username** *username*	(Optional) Specify the remote username.
Step 5	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 6	**archive download-sw /overwrite /reload rcp:**[[[//[*username***@**]*location*]/*directory*]/*image-name***.tar**]	Download the image file from the RCP server to the switch, and overwrite the current image. • The **/overwrite** option overwrites the software image in flash memory with the downloaded image. • The **/reload** option reloads the system after downloading the image unless the configuration has been changed and not been saved. • For //*username*, specify the username. For the RCP copy request to execute successfully, an account must be defined on the network server for the remote username. For more information, see the "Preparing to Download or Upload an Image File By Using RCP" section on page B-28. • For **@***location*, specify the IP address of the RCP server. • For /*directory*/*image-name***.tar**, specify the directory (optional) and the image to download. Directory and image names are case sensitive.
Step 7	**archive download-sw /leave-old-sw /reload rcp:**[[[//[*username***@**]*location*]/*directory*]/*image-name***.tar**]	Download the image file from the RCP server to the switch, and keep the current image. • The **/leave-old-sw** option keeps the old software version after a download. • The **/reload** option reloads the system after downloading the image unless the configuration has been changed and not been saved. • For //*username*, specify the username. For the RCP copy request to execute, an account must be defined on the network server for the remote username. For more information, see the "Preparing to Download or Upload an Image File By Using RCP" section on page B-28. • For **@***location*, specify the IP address of the RCP server. • For /*directory*]/*image-name***.tar**, specify the directory (optional) and the image to download. Directory and image names are case sensitive.

The download algorithm verifies that the image is appropriate for the switch model and that enough DRAM is present, or it aborts the process and reports an error. If you specify the **/overwrite** option, the download algorithm removes the existing image on the flash device whether or not it is the same as the new one, downloads the new image, and then reloads the software.

Note If the flash device has sufficient space to hold two images and you want to overwrite one of these images with the same version, you must specify the **/overwrite** option.

If you specify the **/leave-old-sw**, the existing files are not removed. If there is not enough room to install the new image an keep the running image, the download process stops, and an error message is displayed.

The algorithm installs the downloaded image onto the system board flash device (flash:). The image is placed into a new directory named with the software version string, and the BOOT environment variable is updated to point to the newly installed image.

If you kept the old software during the download process (you specified the **/leave-old-sw** keyword), you can remove it by entering the **delete /force /recursive** *filesystem*:/*file-url* privileged EXEC command. For *filesystem*, use **flash:** for the system board flash device. For *file-url*, enter the directory name of the old software image. All the files in the directory and the directory are removed.

Caution For the download and upload algorithms to operate properly, do *not* rename image names.

Uploading an Image File By Using RCP

You can upload an image from the switch to an RCP server. You can later download this image to the same switch or to another switch of the same type.

The upload feature should be used only if the web management pages associated with the embedded device manager have been installed with the existing image.

Beginning in privileged EXEC mode, follow these steps to upload an image to an RCP server:

	Command	Purpose
Step 1		Verify that the RCP server is properly configured by referring to the "Preparing to Download or Upload an Image File By Using RCP" section on page B-28.
Step 2		Log into the switch through the console port or a Telnet session.
Step 3	**configure terminal**	Enter global configuration mode. This step is required only if you override the default remote username (see Steps 4 and 5).
Step 4	**ip rcmd remote-username** *username*	(Optional) Specify the remote username.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**archive upload-sw rcp:**[[[//[*username***@**]*location*]/*directory*]/*image-name***.tar**]	Upload the currently running switch image to the RCP server. • For //*username*, specify the username; for the RCP copy request to execute, an account must be defined on the network server for the remote username. For more information, see the "Preparing to Download or Upload an Image File By Using RCP" section on page B-28. • For **@***location*, specify the IP address of the RCP server. • For /*directory*]/*image-name***.tar**, specify the directory (optional) and the name of the software image to be uploaded. Directory and image names are case sensitive. • The *image-name***.tar** is the name of software image to be stored on the server.

The **archive upload-sw** privileged EXEC command builds an image file on the server by uploading these files in order: info, the Cisco IOS image, and the web management files. After these files are uploaded, the upload algorithm creates the tar file format.

Caution For the download and upload algorithms to operate properly, do *not* rename image names.



Unsupported Commands in Cisco IOS Release 12.2(25)EX

This appendix lists some of the command-line interface (CLI) commands that appear when you enter the question mark (?) at the Cisco Metro Ethernet (ME) 3400 Ethernet Access switch prompt but are not supported in this release, either because they are not tested or because of switch hardware limitations. This is not a complete list. The unsupported commands are listed by software feature and command mode.

Access Control Lists

Unsupported Privileged EXEC Commands

access-enable [**host**] [**timeout** *minutes*]

access-template [*access-list-number* | *name*] [*dynamic-name*] [*source*] [*destination*] [**timeout** *minutes*]

clear access-template [*access-list-number* | *name*] [*dynamic-name*] [*source*] [*destination*].

Unsupported Global Configuration Commands

access-list rate-limit *acl-index* {*precedence* | **mask** *prec-mask*}

access-list dynamic extended

ARP Commands

Unsupported Global Configuration Commands

arp *ip-address hardware-address* **smds**

arp *ip-address hardware-address* **srp-a**

arp *ip-address hardware-address* **srp-b**

Unsupported Interface Configuration Commands

arp probe

ip probe proxy

Unsupported Debug Commands

debug platform cli-redirection main

debug platform configuration

HSRP

Unsupported Global Configuration Commands

interface Async

interface BVI

interface Dialer

interface Group-Async

interface Lex

interface Multilink

interface Virtual-Template

interface Virtual-Tokenring

Unsupported Interface Configuration Commands

mtu

standby mac-refresh *seconds*

standby use-bia

IGMP Snooping Commands

Unsupported Global Configuration Commands

ip igmp snooping source-only-learning

Interface Commands

Unsupported Privileged EXEC Commands

show interfaces [*interface-id* | **vlan** *vlan-id*] [**crb** | **fair-queue** | **irb** | **mac-accounting** | **precedence** | **irb** | **random-detect** | **rate-limit** | **shape**]

Unsupported Global Configuration Commands

interface tunnel

Unsupported Interface Configuration Commands

transmit-interface *type number*

IP Multicast Routing

Unsupported Privileged EXEC Commands

clear ip rtp header-compression [*type number*]

clear ip dvmrp route commands

debug ip dvmrp commands

The **debug ip packet** command displays packets received by the switch CPU. It does not display packets that are hardware-switched.

The **debug ip mcache** command affects packets received by the switch CPU. It does not display packets that are hardware-switched.

The **debug ip mpacket** [**detail**] [*access-list-number* [*group-name-or-address*] command affects only packets received by the switch CPU. Because most multicast packets are hardware-switched, use this command only when you know that the route will forward the packet to the CPU.

debug ip pim atm

show frame-relay ip rtp header-compression [**interface** *type number*]

show ip dvmrp route commands

The **show ip mcache** command displays entries in the cache for those packets that are sent to the switch CPU. Because most multicast packets are switched in hardware without CPU involvement, you can use this command, but multicast packet information is not displayed.

The **show ip mpacket** commands are supported but are only useful for packets received at the switch CPU. If the route is hardware-switched, the command has no effect because the CPU does not receive the packet and cannot display it.

show ip pim vc [*group-address | name*] [*type number*]

show ip rtp header-compression [*type number*] [**detail**]

Unsupported Global Configuration Commands

All **ip dvmrp** commands

ip pim accept-rp {*address* | **auto-rp**} [*group-access-list-number*]

ip pim message-interval *seconds*

ip pim register-rate-limit

Unsupported Interface Configuration Commands

frame-relay ip rtp header-compression [**active** | **passive**]

frame-relay map ip *ip-address dlci* [**broadcast**] **compress**

frame-relay map ip *ip-address dlci* **rtp header-compression** [**active** | **passive**]

All **ip dvmrp** commands

ip igmp helper-address *ip-address*

ip multicast helper-map {*group-address* | **broadcast**} {*broadcast-address* | *multicast-address*} *extended-access-list-number*

ip multicast rate-limit {**in** | **out**} [**video** | **whiteboard**] [**group-list** *access-list*] [**source-list** *access-list*] *kbps*

ip multicast ttl-threshold *ttl-value* (instead, use the **ip multicast boundary** *access-list-number* interface configuration command)

ip multicast use-functional

ip pim minimum-vc-rate *pps*

ip pim multipoint-signalling

ip pim nbma-mode

ip pim vc-count *number*

ip rtp compression-connections *number*

ip rtp header-compression [**passive**]

IP Unicast Routing

Unsupported Privileged EXEC or User EXEC Commands

clear ip accounting [**checkpoint**]

clear ip bgp *address* flap-statistics

clear ip bgp prefix-list

debug ip cef stats

show cef [**drop** | **not-cef-switched**]

show ip accounting [**checkpoint**] [**output-packets** | **access-violations**]

show ip bgp dampened-paths

show ip bgp inconsistent-as

show ip bgp regexp *regular expression*

show ip prefix-list *regular expression*

Unsupported Global Configuration Commands

ip accounting-list ***ip-address wildcard***

ip as-path access-list

ip accounting-transits *count*

ip cef accounting [**per-prefix**] [**non-recursive**]

ip cef traffic-statistics [**load-interval** *seconds*] [**update-rate** *seconds*]]

ip flow-aggregation

ip flow-cache

ip flow-export

ip gratuitous-arps

ip local

ip prefix-list

ip reflexive-list

router egp

router-isis

router iso-igrp

router mobile

router odr

router static

Unsupported Interface Configuration Commands

ip accounting

ip load-sharing [**per-packet**]

ip mtu *bytes*

ip verify

ip unnumbered *type number*

All **ip security** commands

Unsupported BGP Router Configuration Commands

address-family vpnv4

default-information originate

neighbor advertise-map

neighbor allowas-in

neighbor default-originate

neighbor description

network backdoor

table-map

Unsupported VPN Configuration Commands

All

Unsupported Route Map Commands

match route-type

set as-path {**tag | prepend** *as-path-string*}

set automatic-tag

set dampening *half-life reuse suppress max-suppress-time*

set default interface *interface-id* [*interface-id.....*]

set interface *interface-id* [*interface-id.....*]

set ip default next-hop *ip-address* [*ip-address.....*]

set ip destination *ip-address mask*

set ip precedence *value*

set ip *qos-group*

set metric-type internal

set origin

set metric-type internal

set tag *tag-value*

MAC Address Commands

Unsupported Privileged EXEC Commands

show mac-address-table

show mac-address-table address

show mac-address-table aging-time

show mac-address-table count

show mac-address-table dynamic

show mac-address-table interface

show mac-address-table multicaset

show mac-address-table notification

show mac-address-table static

show mac-address-table vlan

show mac address-table multicast

Note Use the **show ip igmp snooping groups** privileged EXEC command to display Layer 2 multicast address-table entries for a VLAN.

Unsupported Global Configuration Commands

mac-address-table aging-time

mac-address-table notification

mac-address-table static

Miscellaneous

Unsupported Global Configuration Commands

errdisable detect cause dhcp-rate-limit

errdisable recovery cause dhcp-rate-limit

errdisable recovery cause unicast flood

l2protocol-tunnel global drop-threshold

service compress-config

stack-mac persistent timer

Unsupported Privileged EXEC Commands

archive config

remote command all

show archive config

show archive log

show cable-diagnostics prbs

test cable-diagnostics prbs

Unsupported show platform Commands

show platform ipv6 unicast

show platform tb

MSDP

Unsupported Privileged EXEC Commands

show access-expression

show exception

show location

show pm LINE

show smf [*interface-id*]

show subscriber-policy [*policy-number*]

show template [*template-name*]

Unsupported Global Configuration Commands

ip msdp default-peer *ip-address* | *name* [**prefix-list** *list*] (Because BGP/MBGP is not supported, use the **ip msdp peer** command instead of this command.)

NetFlow Commands

Unsupported Global Configuration Commands

ip flow-aggregation cache

ip flow-cache entries

ip flow-export

QoS

Unsupported Global Configuration Commands

priority-list

Unsupported Interface Configuration Commands

priority-group

RADIUS

Unsupported Global Configuration Commands

aaa nas port extended

radius-server attribute nas-port

radius-server configure

radius-server extended-portnames

SNMP

Unsupported Global Configuration Commands

snmp-server enable informs

snmp-server ifindex persist

Spanning Tree

Unsupported Global Configuration Command

spanning-tree pathcost method {long | short}

Unsupported Interface Configuration Command

spanning-tree stack-port

VLAN

Unsupported Global Configuration Commands

vlan internal allocation policy {ascending | descending}

Unsupported User EXEC Commands

show running-config vlan

show vlan ifindex



A

abbreviating commands 2-3
ABRs 32-24
access-class command 28-19
access control entries
 See ACEs
access-denied response, VMPS 11-23
access groups
 applying IPv4 ACLs to interfaces 28-20
 Layer 2 28-20
 Layer 3 28-20
access lists
 See ACLs
access ports
 and Layer 2 protocol tunneling 13-11
 defined 9-3
accounting
 with 802.1x 8-5, 8-18
 with RADIUS 7-28
 with TACACS+ 7-11, 7-17
ACEs
 defined 28-2
 Ethernet 28-2
 IP 28-2
ACL classification, QoS 30-10
ACLs
 ACEs 28-2
 any keyword 28-12, 30-29
ACLs (continued)
 applying
 on multicast packets 28-39
 on routed packets 28-38
 on switched packets 28-37
 time ranges to 28-16
 to an interface 28-19
 to QoS 30-10
 classifying traffic for QoS 30-27
 comments in 28-18
 compiling 28-21
 defined 28-1, 28-7
 examples of 28-21
 extended IPv4
 creating 28-10
 matching criteria 28-7
 hardware and software handling 28-21
 host keyword 28-12, 30-29
 IP
 creating 28-7
 implicit deny 28-9, 28-13, 28-15
 implicit masks 28-9
 matching criteria 28-7
 undefined 28-20
 IPv4
 applying to interfaces 28-19
 creating 28-7
 matching criteria 28-7
 named 28-14
 numbers 28-8
 terminal lines, setting on 28-18
 unsupported features 28-7
 Layer 4 information in 28-37

ACLs (continued)
logging messages 28-8
MAC extended 28-26
matching 28-7, 28-20
monitoring 28-39
named, IPv4 28-14
port 28-2
precedence of 28-2
QoS 30-10
resequencing entries 28-14
router 28-2
router ACLs and VLAN map configuration guidelines 28-36
standard IPv4
creating 28-9
matching criteria 28-7
support for 1-6
support in hardware 28-21
time ranges 28-16
types supported 28-2
unsupported features, IPv4 28-7
using router ACLs with VLAN maps 28-36
VLAN maps
configuration guidelines 28-30
configuring 28-29
active links 17-1
active router 33-1
address aliasing 20-2
addresses
displaying the MAC address table 5-28
dynamic
accelerated aging 14-9
changing the aging time 5-21
default aging 14-9
defined 5-19
learning 5-20
removing 5-22
MAC, discovering 5-28
addresses (continued)
multicast
group address range 34-2
STP address management 14-8
static
adding and removing 5-24
defined 5-19
address resolution 5-28, 32-7
Address Resolution Protocol
See ARP
adjacency tables, with CEF 32-69
administrative distances
defined 32-81
OSPF 32-29
routing protocol defaults 32-71
advertisements
CDP 22-1
RIP 32-18
aggregate addresses, BGP 32-54
aggregated ports
See EtherChannel
aggregate policers
configuration guidelines 30-39
configuring 30-39
described 30-14
aging, accelerating 14-9
aging time
accelerated
for MSTP 15-20
for STP 14-9, 14-21
MAC address table 5-21
maximum
for MSTP 15-21
for STP 14-21
alarms, RMON 25-3
allowed-VLAN list 11-17
area border routers
See ABRs

ARP
configuring 32-8
defined 1-3, 5-28, 32-8
encapsulation 32-9
static cache configuration 32-8
table
address resolution 5-28
managing 5-28
ASBRs 32-24
AS-path filters, BGP 32-49
assured forwarding, DSCP 30-8
asymmetrical links, and IEEE 802.1Q tunneling 13-4
attributes, RADIUS
vendor-proprietary 7-31
vendor-specific 7-29
audience xxxiii
authentication
EIGRP 32-37
HSRP 33-9
local mode with AAA 7-36
NTP associations 5-5
RADIUS
key 7-21
login 7-23
TACACS+
defined 7-11
key 7-13
login 7-14
See also port-based authentication
authentication keys, and routing protocols 32-82
authoritative time source, described 5-2
authorization
with RADIUS 7-27
with TACACS+ 7-11, 7-16
authorized ports with 802.1x 8-4
autoconfiguration 3-3
autonegotiation
duplex mode 1-2
interface configuration guidelines 9-15
mismatches 36-8
autonomous system boundary routers
See ASBRs
autonomous systems, in BGP 32-43
Auto-RP, described 34-4
autosensing, port speed 1-2
availability, features 1-4

B

backup interfaces
See Flex Links
backup links 17-1
bandwidth, QoS, configuring 30-44
bandwidth command
for CBWFQ 30-18
QoS, configuring 30-44
QoS, described 30-21
with police command 30-22
bandwidth remaining percent command 30-23
banners
configuring
login 5-19
message-of-the-day login 5-18
default configuration 5-17
when displayed 5-17
best-effort packet delivery 30-1
BGP
aggregate addresses 32-54
aggregate routes, configuring 32-54
CIDR 32-54
clear commands 32-58
community filtering 32-51
configuring neighbors 32-53
default configuration 32-40
described 32-39

BGP (continued)
enabling 32-42
monitoring 32-58
multipath support 32-46
neighbors, types of 32-42
path selection 32-46
peers, configuring 32-53
prefix filtering 32-50
resetting sessions 32-45
route dampening 32-57
route maps 32-48
route reflectors 32-56
routing domain confederation 32-55
routing session with multi-VRF CE 32-64
show commands 32-58
supernets 32-54
support for 1-7
Version 4 32-40
binding database
DHCP snooping
See DHCP snooping binding database
bindings
DHCP snooping database 18-5
IP source guard 18-13
binding table, DHCP snooping
See DHCP snooping binding database
blocking packets 21-7
booting
boot loader, function of 3-2
boot process 3-1
manually 3-14
specific image 3-14
boot loader
accessing 3-15
described 3-2
environment variables 3-15
prompt 3-15
trap-door mechanism 3-2
bootstrap router (BSR), described 34-5
Border Gateway Protocol
See BGP
BPDU
error-disabled state 16-2
filtering 16-3
RSTP format 15-9
BPDU filtering
described 16-3
disabling 16-8
enabling 16-7
support for 1-4
BPDU guard
described 16-2
disabling 16-7
enabling 16-6
support for 1-4
broadcast flooding 32-15
broadcast packets
directed 32-12
flooded 32-12
broadcast storm-control command 21-4
broadcast storms 21-1, 32-13

C

cables, monitoring for unidirectional links 23-1
caution, described xxxiv
CBWFQ
and bandwidth command 30-21, 30-44
configuration guidelines 30-44
QoS scheduling 30-18
CDP
configuring 22-2
default configuration 22-2
described 22-1
disabling for routing device 22-3 to 22-4
enabling and disabling
on an interface 22-4
on a switch 22-3

CDP (continued)
Layer 2 protocol tunneling 13-8
monitoring 22-5
overview 22-1
support for 1-3
transmission timer and holdtime, setting 22-2
updates 22-2
CEF
defined 32-69
enabling 32-70
child policies, QoS 30-20
CIDR 32-54
Cisco Configuration Engine 1-3
Cisco Discovery Protocol
See CDP
Cisco Express Forwarding
See CEF
Cisco IOS File System
See IFS
CiscoWorks 2000 1-3, 27-4
class-based priority queuing, QoS 30-15
class-based shaping
configuration guidelines 30-46
configuring 30-46
for QoS 30-19
Class-Based-Weighted-Fair-Queuing
See CBWFQ
classification
based on ACL lookup 30-10
in packet headers 30-5
QoS comparisons 30-9
QoS group 30-10
classless interdomain routing
See CIDR
classless routing 32-6
class map
match-all option 30-7
match-any option 30-7
class-map command 30-3
class maps, QoS
configuring 30-31, 30-32
described 30-6
class of service
See CoS
class selectors, DSCP 30-8
clearing interfaces 9-24
CLI
abbreviating commands 2-3
command modes 2-1
described 1-3
editing features
enabling and disabling 2-6
keystroke editing 2-6
wrapped lines 2-8
error messages 2-4
filtering command output 2-8
getting help 2-3
history
changing the buffer size 2-5
described 2-4
disabling 2-5
recalling commands 2-5
no and default forms of commands 2-4
clock
See system clock
CNS
Configuration Engine
configID, deviceID, hostname 4-3
configuration service 4-2
described 4-1
event service 4-3
embedded agents
described 4-5
enabling automated configuration 4-6
enabling configuration agent 4-9
enabling event agent 4-8
command-line interface
See CLI

command modes 2-1
commands
 abbreviating 2-3
 no and default 2-4
commands, setting privilege levels 7-8
community list, BGP 32-52
community ports 12-3
community strings
 configuring 27-8
 overview 27-4
community VLANs 12-2, 12-3
compatibility, feature 21-11
config.text 3-13
configurable leave timer, IGMP 20-5
configuration, initial
 defaults 1-8
 See also getting started guide and hardware installation guide
configuration examples
 network 1-11
 policy maps 30-56
 QoS
 adding customers 30-58
 adding or deleting a class 30-61
 adding or deleting classification criteria 30-58, 30-59
 adding or deleting configured actions 30-60
 changing queuing or scheduling parameters 30-60
 initial 30-56
configuration files
 clearing the startup configuration B-18
 creating using a text editor B-9
 default name 3-13
 deleting a stored configuration B-18
 described B-8
configuration files (continued)
 downloading
 automatically 3-13
 preparing B-10, B-12, B-15
 reasons for B-8
 using FTP B-13
 using RCP B-16
 using TFTP B-10
 guidelines for creating and using B-8
 invalid combinations when copying B-5
 limiting TFTP server access 27-15
 obtaining with DHCP 3-7
 password recovery disable considerations 7-5
 specifying the filename 3-13
 system contact and location information 27-14
 types and location B-9
 uploading
 preparing B-10, B-12, B-15
 reasons for B-8
 using FTP B-14
 using RCP B-17
 using TFTP B-11
configuration guidelines
 aggregate policers 30-39
 CBWFQ 30-44
 class-based shaping 30-46
 individual policers 30-36
 input policy maps 30-35
 marking in policy maps 30-41
 multi-VRF CE 32-61
 output policy maps 30-43
 QoS, general 30-27
 QoS class maps 30-31
 strict priority queuing 30-48
 unconditional priority policing 30-50
 VLANs 11-8
 WTD 30-53
configuration settings, saving 3-10
configure terminal command 9-7

configuring marking in input policy maps 30-41
congestion avoidance, QoS 30-2, 30-24
congestion management, QoS 30-2, 30-18
connections, secure remote 7-38
connectivity problems 36-10, 36-13, 36-15
console port, connecting to 2-9
conventions
 command xxxiii
 for examples xxxiv
 publication xxxiii
 text xxxiii
corrupted software, recovery steps with Xmodem 36-2
CoS
 classification 30-7
 values 30-5
counters, clearing interface 9-24
crashinfo file 36-22
cryptographic software image
 Kerberos 7-32
 SSH 7-37
customer edge devices 1-14, 32-59

D

daylight saving time 5-13
debugging
 enabling all system diagnostics 36-19
 enabling for a specific feature 36-18
 redirecting error message output 36-19
 using commands 36-18
default actions, table maps 30-11
default commands 2-4
default configuration
 banners 5-17
 BGP 32-40
 booting 3-13
 CDP 22-2
 DHCP 18-7
 DHCP option 82 18-7
default configuration (continued)
 DHCP snooping 18-7
 DHCP snooping binding database 18-7
 DNS 5-16
 dynamic ARP inspection 19-5
 EIGRP 32-34
 EtherChannel 31-9
 Flex Links 17-2
 HSRP 33-4
 IEEE 802.1Q tunneling 13-4
 IEEE 802.1x 8-9
 IGMP 34-25
 IGMP filtering 20-23
 IGMP snooping 20-6
 IGMP throttling 20-23
 initial switch information 3-3
 IP addressing, IP routing 32-4
 IP multicast routing 34-7
 IP source guard 18-14
 Layer 2 interfaces 9-12
 Layer 2 protocol tunneling 13-11
 MAC address table 5-21
 MSDP 35-4
 MSTP 15-12
 multi-VRF CE 32-61
 MVR 20-18
 NTP 5-4
 optional spanning-tree configuration 16-5
 OSPF 32-24
 password and privilege level 7-2
 PIM 34-7
 private VLANs 12-6
 QoS 30-27
 RADIUS 7-20
 RIP 32-19
 RMON 25-3
 RSPAN 24-10
 SDM template 6-2
 SNMP 27-6

default configuration (continued)
SPAN 24-10
STP 14-11
system message logging 26-3
system name and prompt 5-15
TACACS+ 7-13
UDLD 23-4
VLAN, Layer 2 Ethernet interfaces 11-16
VLANs 11-7
VMPS 11-24
default gateway 3-10, 32-11
default networks 32-72
default routes 32-72
default routing 32-2
default service, DSCP 30-8
default template 6-1
description command 9-19
designing your network, examples 1-11
destination addresses, in IPv4 ACLs 28-11
destination-IP address-based forwarding, EtherChannel 31-8
destination-MAC address forwarding, EtherChannel 31-7
device discovery protocol 22-1
DHCP, enabling the relay agent 18-8
DHCP-based autoconfiguration
client request message exchange 3-4
configuring
client side 3-3
DNS 3-6
relay device 3-6
server side 3-5
TFTP server 3-5
example 3-8
lease options
for IP address information 3-5
for receiving the configuration file 3-5
overview 3-3
relationship to BOOTP 3-3
DHCP-based autoconfiguration (continued)
relay support 1-3, 1-8
support for 1-3
DHCP binding database
See DHCP snooping binding database
DHCP binding table
See DHCP snooping binding database
DHCP option 82
circuit ID suboption 18-5
configuration guidelines 18-7
default configuration 18-7
displaying 18-13
forwarding address, specifying 18-9
helper address 18-9
overview 18-3
packet format, suboption
circuit ID 18-5
remote ID 18-5
remote ID suboption 18-5
DHCP snooping
accepting untrusted packets form edge switch 18-3, 18-10
and private VLANs 18-11
binding database
See DHCP snooping binding database
configuration guidelines 18-7
default configuration 18-7
displaying binding tables 18-13
message exchange process 18-4
option 82 data insertion 18-3
trusted interface 18-2
untrusted interface 18-2
untrusted messages 18-2
DHCP snooping binding database
adding bindings 18-12
binding entries, displaying 18-13
binding file
format 18-6
location 18-5

DHCP snooping binding database (continued)
bindings 18-5
clearing agent statistics 18-12
configuration guidelines 18-8
configuring 18-12
default configuration 18-7
deleting
binding file 18-12
bindings 18-12
database agent 18-12
described 18-5
displaying 18-13
binding entries 18-13
status and statistics 18-13
enabling 18-12
entry 18-5
renewing database 18-12
resetting
delay value 18-12
timeout value 18-12
DHCP snooping binding table
See DHCP snooping binding database
Differentiated Services Code Point
See DSCP
Diffusing Update Algorithm (DUAL) 32-32
directed unicast requests 1-3
directories
changing B-3
creating and removing B-4
displaying the working B-3
distance-vector protocols 32-2
distribute-list command 32-80
DNS
and DHCP-based autoconfiguration 3-6
default configuration 5-16
displaying the configuration 5-17
overview 5-15
setting up 5-16
support for 1-3
documentation, related xxxiv
document conventions xxxiii
domain names
DNS 5-15
Domain Name System
See DNS
dot1q-tunnel switchport mode 11-15
double-tagged packets
IEEE 802.1Q tunneling 13-2
Layer 2 protocol tunneling 13-10
downloading
configuration files
preparing B-10, B-12, B-15
reasons for B-8
using FTP B-13
using RCP B-16
using TFTP B-10
image files
deleting old image B-22
preparing B-21, B-24, B-28
reasons for B-18
using FTP B-25
using RCP B-29
using TFTP B-21
using the device manager or Network Assistant B-18
drop threshold for Layer 2 protocol packets 13-11
DSCP
assured forwarding 30-8
classification 30-8
class selectors 30-8
default service 30-8
expedited forwarding 30-8
values 30-5
DUAL finite state machine, EIGRP 32-33
duplex mode, configuring 9-15
dynamic access ports
characteristics 11-5
configuring 11-26
defined 9-3

dynamic addresses
See addresses
dynamic ARP inspection
ARP cache poisoning 19-1
ARP requests, described 19-1
ARP spoofing attack 19-1
clearing
log buffer 19-15
statistics 19-15
configuration guidelines 19-6
configuring
ACLs for non-DHCP environments 19-8
in DHCP environments 19-7
log buffer 19-13
rate limit for incoming ARP packets 19-4, 19-10
default configuration 19-5
denial-of-service attacks, preventing 19-10
described 19-1
DHCP snooping binding database 19-2
displaying
ARP ACLs 19-14
configuration and operating state 19-14
log buffer 19-15
statistics 19-15
trust state and rate limit 19-14
error-disabled state for exceeding rate limit 19-4
function of 19-2
interface trust states 19-3
log buffer
clearing 19-15
configuring 19-13
displaying 19-15
logging of dropped packets, described 19-4
man-in-the middle attack, described 19-2
network security issues and interface trust states 19-3
priority of ARP ACLs and DHCP snooping entries 19-4
dynamic ARP inspection (continued)
rate limiting of ARP packets
configuring 19-10
described 19-4
error-disabled state 19-4
statistics
clearing 19-15
displaying 19-15
validation checks, performing 19-12
Dynamic Host Configuration Protocol
See DHCP-based autoconfiguration
dynamic port VLAN membership
described 11-24
reconfirming 11-26, 11-27
troubleshooting 11-28
types of connections 11-26
dynamic routing 32-2

E

EBGP 32-38
editing features
enabling and disabling 2-6
keystrokes used 2-6
wrapped lines 2-8
EIGRP
authentication 32-37
components 32-33
configuring 32-35
default configuration 32-34
definition 32-32
interface parameters, configuring 32-36
monitoring 32-38
support for 1-7
enable password 7-4
enable secret password 7-4
encryption for passwords 7-4
Enhanced IGRP
See EIGRP

environment variables, function of 3-16
equal-cost routing 1-7, 32-70
error messages during command entry 2-4
EtherChannel
 802.3ad, described 31-6
 automatic creation of 31-4, 31-6
 channel groups
 binding physical and logical interfaces 31-3
 numbering of 31-3
 configuration guidelines 31-10
 configuring
 Layer 2 interfaces 31-11
 Layer 3 physical interfaces 31-14
 Layer 3 port-channel logical interfaces 31-13
 default configuration 31-9
 described 31-2
 displaying status 31-21
 forwarding methods 31-7, 31-16
 interaction
 with STP 31-10
 with VLANs 31-11
 LACP
 described 31-6
 displaying status 31-21
 hot-standby ports 31-18
 interaction with other features 31-6
 modes 31-6
 port priority 31-20
 system priority 31-19
 Layer 3 interface 32-3
 load balancing 31-7, 31-16
 logical interfaces, described 31-3
EtherChannel (continued)
 PAgP
 aggregate-port learners 31-17
 compatibility with Catalyst 1900 31-17
 described 31-4
 displaying status 31-21
 interaction with other features 31-5
 learn method and priority configuration 31-17
 modes 31-5
 support for 1-2
 port-channel interfaces
 described 31-3
 numbering of 31-3
 port groups 9-6
 support for 1-2
EtherChannel guard
 described 16-3
 disabling 16-9
 enabling 16-8
Ethernet VLANs
 adding 11-9
 defaults and ranges 11-7
 modifying 11-9
events, RMON 25-3
examples
 conventions for xxxiv
 network configuration 1-11
expedited forwarding, DSCP 30-8
extended-range VLANs
 creating with an internal VLAN ID 11-11
 defined 11-1
extended system ID
 MSTP 15-14
 STP 14-4, 14-14
Extensible Authentication Protocol over LAN 8-1
external BGP
 See EBGP
external neighbors, BGP 32-42

F

features, incompatible 21-11
FIB 32-69
fiber-optic, detecting unidirectional links 23-1
files
 copying B-4
 crashinfo
 description 36-22
 displaying the contents of 36-22
 location 36-22
 deleting B-5
 displaying the contents of B-7
 tar
 creating B-6
 displaying the contents of B-6
 extracting B-7
 image file format B-19
file system
 displaying available file systems B-2
 displaying file information B-3
 local file system names B-1
 network file system names B-4
 setting the default B-3
filtering
 in a VLAN 28-29
 non-IP traffic 28-26
 show and more command output 2-8
filtering show and more command output 2-8
filters, IP
 See ACLs, IP
flash device, number of B-1
Flex Links
 configuration guidelines 17-2
 configuring 17-3
 default configuration 17-2
 description 17-1
 monitoring 17-4
flooded traffic, blocking 21-7
flow control 1-2, 9-17
forward-delay time
 MSTP 15-20
 STP 14-21
Forwarding Information Base
 See FIB
FTP
 accessing MIB files A-3
 configuration files
 downloading B-13
 overview B-11
 preparing the server B-12
 uploading B-14
 image files
 deleting old image B-26
 downloading B-25
 preparing the server B-24
 uploading B-27

G

get-bulk-request operation 27-3
get-next-request operation 27-3, 27-4
get-request operation 27-3, 27-4
get-response operation 27-3
global configuration mode 2-2
global leave, IGMP 20-11
guide
 audience xxxiii
 purpose of xxxiii

H

hardware limitations and Layer 3 interfaces 9-20
hello time
 MSTP 15-19
 STP 14-20
help, for the command line 2-3

history
changing the buffer size 2-5
described 2-4
disabling 2-5
recalling commands 2-5
history table, level and number of syslog messages 26-9
host ports
configuring 12-12
kinds of 12-3
hosts, limit on dynamic ports 11-28
Hot Standby Router Protocol
See HSRP
HP OpenView 1-3
HSRP
authentication string 33-9
command-switch redundancy 1-4
configuring 33-4
default configuration 33-4
definition 33-1
guidelines 33-5
monitoring 33-11
overview 33-1
priority 33-6
routing redundancy 1-7
support for ICMP redirect messages 33-11
timers 33-9
tracking 33-7

I

IBPG 32-38
ICMP
redirect messages 32-11
support for 1-7
time-exceeded messages 36-15
traceroute and 36-15
unreachable messages 28-19
unreachables and ACLs 28-21
ICMP ping
executing 36-10
overview 36-10
ICMP Router Discovery Protocol
See IRDP
IDS appliances
and ingress RSPAN 24-20
and ingress SPAN 24-13
IEEE 802.1D
See STP
IEEE 802.1Q
and trunk ports 9-4
configuration limitations 11-15
encapsulation 11-14
native VLAN for untagged traffic 11-19
tunneling
compatibility with other features 13-6
defaults 13-4
described 13-1
tunnel ports with other features 13-6
IEEE 802.1s
See MSTP
IEEE 802.1w
See RSTP
IEEE 802.1x
See port-based authentication
IEEE 802.3ad
See EtherChannel
IEEE 802.3z flow control 9-17
ifIndex values, SNMP 27-5
IFS 1-4
IGMP
configurable leave timer, described 20-5
configurable leave timer, procedures 20-10
configuring the switch
as a member of a group 34-25
statically connected member 34-30
controlling access to groups 34-26
default configuration 34-25

IGMP (continued)
deleting cache entries 34-34
displaying groups 34-35
fast switching 34-30
flooded multicast traffic
controlling flooding time 20-11
disabling on an interface 20-12
global leave 20-11
query solicitation 20-11
recovering from flood mode 20-11
host-query interval, modifying 34-28
joining multicast group 20-3
join messages 20-3
leave processing, enabling 20-9
leaving multicast group 20-5
multicast reachability 34-25
overview 34-2
queries 20-3
report suppression
described 20-6
disabling 20-14
supported versions 20-2
support for 1-2
Version 1
changing to Version 2 34-27
described 34-3
Version 2
changing to Version 1 34-27
described 34-3
maximum query response time value 34-29
pruning groups 34-29
query timeout value 34-29
IGMP configurable leave timer 20-5
IGMP filtering
configuring 20-23
default configuration 20-23
described 20-22
monitoring 20-28
support for 1-3
IGMP groups
configuring filtering 20-26
setting the maximum number 20-25
IGMP leave timer, configuring 20-10
IGMP profile
applying 20-25
configuration mode 20-23
configuring 20-24
IGMP snooping
and address aliasing 20-2
configuring 20-6
default configuration 20-6
definition 20-1
enabling and disabling 20-7
global configuration 20-7
Immediate Leave 20-5
monitoring 20-15
querier
configuration guidelines 20-13
configuring 20-13
supported versions 20-2
support for 1-2
VLAN configuration 20-7
IGMP throttling
configuring 20-26
default configuration 20-23
described 20-23
displaying action 20-28
IGP 32-23
Immediate Leave, IGMP
configuration guidelines 20-10
described 20-5
enabling 20-9
individual policers
configuration guidelines 30-36
configuring 30-36
initial configuration
defaults 1-8
See also getting started and hardware installation guides

input policy maps
classification criteria 30-4
configuration guidelines 30-35
configuring 30-35
displaying statistics 30-55
interface
number 9-7
range macros 9-10
interface command 9-7
interface configuration mode 2-2
interfaces
configuration guidelines
duplex and speed 9-15
configuring
duplex mode 9-15
procedure 9-7
speed 9-15
counters, clearing 9-24
described 9-19
descriptive name, adding 9-19
displaying information about 9-23
flow control 9-17
management 1-3
monitoring 9-22
naming 9-19
physical, identifying 9-7
range of 9-8
restarting 9-24
shutting down 9-24
status 9-22
supported 9-7
types of 9-1
interfaces range macro command 9-10
interface types 9-7
NNI 9-3
UNI 9-3
Interior Gateway Protocol
See IGP
internal BGP
See IBGP
internal neighbors, BGP 32-42
Internet Control Message Protocol
See ICMP
Internet Group Management Protocol
See IGMP
inter-VLAN routing 1-7, 32-2
Intrusion Detection System
See IDS appliances
IP ACLs
for QoS classification 30-10
implicit deny 28-9, 28-13
implicit masks 28-9
named 28-14
undefined 28-20
IP addresses
classes of 32-5
default configuration 32-4
discovering 5-28
for IP routing 32-4
MAC address association 32-7
monitoring 32-17
IP broadcast address 32-15
ip cef distributed command 32-69
IP directed broadcasts 32-13
ip igmp profile command 20-23
IP information
assigned
manually 3-9
through DHCP-based autoconfiguration 3-3
default configuration 3-3
IP multicast routing
addresses
all-hosts 34-2
all-multicast-routers 34-2
host group address range 34-2
administratively-scoped boundaries, described 34-32
and IGMP snooping 20-1

IP multicast routing (continued)
Auto-RP
adding to an existing sparse-mode cloud 34-13
benefits of 34-12
clearing the cache 34-34
configuration guidelines 34-8
filtering incoming RP announcement messages 34-15
overview 34-4
preventing candidate RP spoofing 34-15
preventing join messages to false RPs 34-14
setting up in a new internetwork 34-12
using with BSR 34-20
bootstrap router
configuration guidelines 34-8
configuring candidate BSRs 34-18
configuring candidate RPs 34-19
defining the IP multicast boundary 34-17
defining the PIM domain border 34-16
overview 34-5
using with Auto-RP 34-20
Cisco implementation 34-2
configuring
basic multicast routing 34-9
IP multicast boundary 34-32
default configuration 34-7
enabling
multicast forwarding 34-9
PIM mode 34-10
group-to-RP mappings
Auto-RP 34-4
BSR 34-5
IP multicast routing (continued)
MBONE
deleting sdr cache entries 34-34
described 34-31
displaying sdr cache 34-35
enabling sdr listener support 34-31
limiting sdr cache entry lifetime 34-32
SAP packets for conference session announcement 34-31
Session Directory (sdr) tool, described 34-31
monitoring
packet rate loss 34-35
peering devices 34-35
tracing a path 34-35
multicast forwarding, described 34-5
PIMv1 and PIMv2 interoperability 34-8
reverse path check (RPF) 34-5
routing table
deleting 34-34
displaying 34-35
RP
assigning manually 34-11
configuring Auto-RP 34-12
configuring PIMv2 BSR 34-16
monitoring mapping information 34-21
using Auto-RP and BSR 34-20
statistics, displaying system and network 34-34
See also IGMP
See also PIM
IP packets, classification 30-5
IP precedence
classification 30-8
values 30-5
IP protocols
in ACLs 28-11
routing 1-7
IP routes, monitoring 32-83

IP routing
connecting interfaces with 9-7
disabling 32-18
enabling 32-17
IP source guard
and 802.1x 18-15
and DHCP snooping 18-13
and EtherChannels 18-15
and port security 18-15
and private VLANs 18-15
and routed ports 18-15
and TCAM entries 18-15
and trunk interfaces 18-15
and VRF 18-15
binding configuration
automatic 18-13
manual 18-13
binding table 18-13
configuration guidelines 18-15
default configuration 18-14
described 18-13
disabling 18-16
displaying
bindings 18-16
configuration 18-16
enabling 18-15
filtering
source IP address 18-14
source IP and MAC address 18-14
source IP address filtering 18-14
source IP and MAC address filtering 18-14
static bindings
adding 18-15
deleting 18-16
IP traceroute
executing 36-16
overview 36-15
IP unicast routing
address resolution 32-7
administrative distances 32-71, 32-81
ARP 32-8
assigning IP addresses to Layer 3 interfaces 32-5
authentication keys 32-82
broadcast
address 32-15
flooding 32-15
packets 32-12
storms 32-13
classless routing 32-6
configuring static routes 32-71
default
addressing configuration 32-4
gateways 32-11
networks 32-72
routes 32-72
routing 32-2
directed broadcasts 32-13
disabling 32-18
dynamic routing 32-2
enabling 32-17
EtherChannel Layer 3 interface 32-3
IGP 32-23
inter-VLAN 32-2
IP addressing
classes 32-5
configuring 32-4
IRDP 32-11
Layer 3 interfaces 32-3
MAC address and IP address 32-7
passive interfaces 32-79
protocols
distance-vector 32-2
dynamic 32-2
link-state 32-3
proxy ARP 32-8
redistribution 32-73

IP unicast routing (continued)
reverse address resolution 32-7
routed ports 32-3
static routing 32-2
steps to configure 32-3
subnet mask 32-5
subnet zero 32-6
supernet 32-6
UDP 32-14
with SVIs 32-3
See also BGP
See also EIGRP
See also OSPF
See also RIP
IPv4 ACLs
applying to interfaces 28-19
extended, creating 28-10
named 28-14
standard, creating 28-9
IRDP
configuring 32-11
definition 32-11
support for 1-7
ISL trunking with IEEE 802.1 tunneling 13-5
isolated port 12-3
isolated VLANs 12-2, 12-3

J

join messages, IGMP 20-3

K

KDC
described 7-32
See also Kerberos
keepalive messages 14-3
Kerberos
authenticating to
boundary switch 7-35
KDC 7-35
network services 7-35
configuration examples 7-32
configuring 7-36
credentials 7-32
cryptographic software image 7-32
described 7-32
KDC 7-32
operation 7-34
realm 7-33
server 7-33
support for 1-6
switch as trusted third party 7-32
terms 7-33
TGT 7-34
tickets 7-32
key distribution center
See KDC

L

l2protocol-tunnel command 13-13
LACP
Layer 2 protocol tunneling 13-9
See EtherChannel
Layer 2 interfaces, default configuration 9-12
Layer 2 packets, classification 30-5
Layer 2 protocol tunneling
configuring 13-10
configuring for EtherChannels 13-14
default configuration 13-11
defined 13-8
guidelines 13-11
layer-2 template 6-1

Layer 2 traceroute
and ARP 36-14
and CDP 36-14
broadcast traffic 36-13
described 36-13
IP addresses and subnets 36-14
MAC addresses and VLANs 36-14
multicast traffic 36-14
multiple devices on a port 36-14
unicast traffic 36-13
usage guidelines 36-14
Layer 3 features 1-7
Layer 3 interfaces
assigning IP addresses to 32-5
changing from Layer 2 mode 32-5
types of 32-3
LDAP 4-2
lightweight directory access protocol
See LDAP
line configuration mode 2-2
Link Aggregation Control Protocol
See EtherChannel
link redundancy
See Flex Links
links, unidirectional 23-1
link state advertisements (LSAs) 32-27
link-state protocols 32-3
load balancing 33-3
local SPAN 24-2
logging messages, ACL 28-8
login authentication
with RADIUS 7-23
with TACACS+ 7-14
login banners 5-17
log messages
See system message logging
loop guard
described 16-4
enabling 16-9
support for 1-4

M

MAC addresses
aging time 5-21
and VLAN association 5-20
building the address table 5-20
default configuration 5-21
disabling learning on a VLAN 5-27
discovering 5-28
displaying 5-28
displaying in the IP source binding table 18-16
dynamic
learning 5-20
removing 5-22
in ACLs 28-26
IP address association 32-7
static
adding 5-25
allowing 5-26, 5-27
characteristics of 5-24
dropping 5-26
removing 5-25
MAC address learning, disabling on a VLAN 5-27
MAC address notification, support for 1-8
MAC address-to-VLAN mapping 11-23
MAC extended access lists
applying to Layer 2 interfaces 28-28
configuring for QoS 30-30
creating 28-26
defined 28-26
macros
See Smartports macros
manageability features 1-3

management access
in-band
CLI session 1-4
SNMP 1-4
out-of-band console port connection 1-4
management options
CLI 2-1
CNS 4-1
overview 1-3
marking
action with aggregate policers 30-39
described 30-2, 30-12
match command, QoS
for classification 30-3, 30-7
guidelines 30-31
matching, IPv4 ACLs 28-7
matching classifications, QoS 30-7
maximum aging time
MSTP 15-21
STP 14-21
maximum hop count, MSTP 15-21
maximum-paths command 32-46, 32-70
membership mode, VLAN port 11-4
messages
to users through banners 5-17
messages, to users through banners 5-17
metrics, in BGP 32-46
metric translations, between routing protocols 32-76
metro tags 13-2
MHSRP 33-3
MIBs
accessing files with FTP A-3
location of files A-3
overview 27-1
SNMP interaction with 27-4
supported A-1
mirroring traffic for analysis 24-1
mismatches, autonegotiation 36-8
modular QoS command-line interface
See MQC
module number 9-7
monitoring
access groups 28-39
BGP 32-58
cables for unidirectional links 23-1
CDP 22-5
CEF 32-70
EIGRP 32-38
features 1-8
Flex Links 17-4
HSRP 33-11
IEEE 802.1Q tunneling 13-18
IGMP
filters 20-28
snooping 20-15
interfaces 9-22
IP
address tables 32-17
multicast routing 34-34
routes 32-83
IPv4 ACL configuration 28-39
Layer 2 protocol tunneling 13-18
MSDP peers 35-19
multicast router interfaces 20-15
multi-VRF CE 32-68
MVR 20-22
network traffic for analysis with probe 24-2
OSPF 32-31
port
blocking 21-17
protection 21-17
private VLANs 12-15
QoS 30-55
RP mapping information 34-21
SFP status 1-8, 9-23, 36-9
source-active messages 35-19
speed and duplex mode 9-16

monitoring (continued)
traffic flowing among switches 25-1
traffic suppression 21-17
tunneling 13-18
VLAN
filters 28-40
maps 28-40
VLANs 11-14
VMPS 11-27
MQC
process 30-3
steps to configure 30-3
MSDP
benefits of 35-3
clearing MSDP connections and statistics 35-19
controlling source information
forwarded by switch 35-12
originated by switch 35-8
received by switch 35-14
default configuration 35-4
dense-mode regions
sending SA messages to 35-17
specifying the originating address 35-18
filtering
incoming SA messages 35-14
SA messages to a peer 35-12
SA requests from a peer 35-11
join latency, defined 35-6
meshed groups
configuring 35-16
defined 35-16
originating address, changing 35-18
overview 35-1
peer-RPF flooding 35-2
MSDP (continued)
peers
configuring a default 35-4
monitoring 35-19
peering relationship, overview 35-1
requesting source information from 35-8
shutting down 35-16
source-active messages
caching 35-6
clearing cache entries 35-19
defined 35-2
filtering from a peer 35-11
filtering incoming 35-14
filtering to a peer 35-12
limiting data with TTL 35-14
monitoring 35-19
restricting advertised sources 35-9
support for 1-8
MSTP
boundary ports
configuration guidelines 15-12
described 15-5
BPDU filtering
described 16-3
enabling 16-7
BPDU guard
described 16-2
enabling 16-6
CIST, described 15-3
configuration guidelines 15-12, 16-5
configuring
forward-delay time 15-20
hello time 15-19
link type for rapid convergence 15-22
maximum aging time 15-21
maximum hop count 15-21
MST region 15-13
path cost 15-18
port priority 15-17

MSTP, configuring (continued)
root switch 15-14
secondary root switch 15-16
switch priority 15-19
CST
defined 15-3
operations between regions 15-3
default configuration 15-12
default optional feature configuration 16-5
displaying status 15-23
enabling the mode 15-13
EtherChannel guard
described 16-3
enabling 16-8
extended system ID
effects on root switch 15-14
effects on secondary root switch 15-16
unexpected behavior 15-15
instances supported 14-10
interface state, blocking to forwarding 16-2
interoperability and compatibility among modes 14-10
interoperability with 802.1D
described 15-5
restarting migration process 15-22
IST
defined 15-2
master 15-3
operations within a region 15-3
loop guard
described 16-4
enabling 16-9
mapping VLANs to MST instance 15-13
MST region
CIST 15-3
configuring 15-13
described 15-2
hop-count mechanism 15-4
IST 15-2
supported spanning-tree instances 15-2
MSTP (continued)
optional features supported 1-4
overview 15-2
Port Fast
described 16-2
enabling 16-5
preventing root switch selection 16-3
root guard
described 16-3
enabling 16-9
root switch
configuring 15-15
effects of extended system ID 15-14
unexpected behavior 15-15
shutdown Port Fast-enabled port 16-2
status, displaying 15-23
multicast groups
Immediate Leave 20-5
leaving 20-5
static joins 20-8
multicast packets
ACLs on 28-39
blocking 21-7
multicast router interfaces, monitoring 20-15
multicast router ports, adding 20-8
Multicast Source Discovery Protocol
See MSDP
multicast storm 21-1
multicast storm-control command 21-4
multicast television application 20-16
multicast VLAN 20-16
Multicast VLAN Registration
See MVR
Multiple HSRP
See MHSRP
multiple VPN routing/forwarding in customer edge devices
See multi-VRF CE

multi-VRF CE
configuration example 32-64
configuration guidelines 32-61
configuring 32-61
default configuration 32-61
defined 1-14, 32-59
displaying 32-68
monitoring 32-68
network components 32-61
packet-forwarding process 32-61
support for 1-8
MVR
and address aliasing 20-19
and IGMPv3 20-19
configuration guidelines 20-19
configuring interfaces 20-20
default configuration 20-18
described 20-16
example application 20-16
in the switch stack 20-18
modes 20-20
monitoring 20-22
multicast television application 20-16
setting global parameters 20-19
support for 1-2

N

named IPv4 ACLs 28-14
NameSpace Mapper
See NSM
native VLAN
and IEEE 802.1Q tunneling 13-4
configuring 11-19
default 11-19
neighbor discovery/recovery, EIGRP 32-33
neighbors, BGP 32-53
network management
CDP 22-1
RMON 25-1
SNMP 27-1
network node interface
See NNI
Network Time Protocol
See NTP
NNI
configuring 9-13
described 9-3
no commands 2-4
non-IP traffic filtering 28-26
nontrunking mode 11-15
normal-range VLANs 11-3
configuring 11-6
defined 11-1
no switchport command 9-5
note, described xxxiv
not-so-stubby areas
See NSSA
NSM 4-3
NSSA, OSPF 32-27
NTP
associations
authenticating 5-5
defined 5-2
enabling broadcast messages 5-7
peer 5-6
server 5-6
default configuration 5-4
displaying the configuration 5-11
overview 5-2
restricting access
creating an access group 5-9
disabling NTP services per interface 5-10
source IP address, configuring 5-10
stratum 5-2
support for 1-3

NTP (continued)
synchronizing devices 5-6
time
services 5-2
synchronizing 5-2

O

Open Shortest Path First
See OSPF
optimizing system resources 6-1
options, management 1-3
OSPF
area parameters, configuring 32-27
configuring 32-26
default configuration
metrics 32-29
route 32-29
settings 32-24
described 32-23
interface parameters, configuring 32-26
LSA group pacing 32-30
monitoring 32-31
router IDs 32-31
route summarization 32-29
support for 1-7
virtual links 32-29
output policies 30-5
output policy maps
classification criteria 30-5
configuration guidelines 30-43
configuring 30-43
displaying statistics 30-55

P

packet classification
defined 30-5
to organize traffic 30-2
packet marking
configuring 30-41
defined 30-16
packet policing, for QoS 30-2
PAgP
Layer 2 protocol tunneling 13-9
See EtherChannel
parallel paths, in routing tables 32-70
parent policies, QoS 30-20
passive interfaces
configuring 32-79
OSPF 32-29
passwords
default configuration 7-2
disabling recovery of 7-5
encrypting 7-4
for security 1-5
overview 7-1
recovery of 36-3
setting
enable 7-3
enable secret 7-4
Telnet 7-6
with usernames 7-7
path cost
MSTP 15-18
STP 14-18
PBR
defined 32-76
enabling 32-78
fast-switched policy-based routing 32-79
local policy-based routing 32-79
peers, BGP 32-53

performance features 1-2
per-VLAN spanning-tree plus
 See PVST+
PE to CE routing, configuring 32-64
physical ports 9-2
PIM
 default configuration 34-7
 dense mode
 overview 34-4
 rendezvous point (RP), described 34-4
 RPF lookups 34-6
 displaying neighbors 34-35
 enabling a mode 34-10
 overview 34-3
 router-query message interval, modifying 34-24
 shared tree and source tree, overview 34-21
 shortest path tree, delaying the use of 34-23
 sparse mode
 join messages and shared tree 34-4
 overview 34-4
 prune messages 34-4
 RPF lookups 34-6
 support for 1-1, 1-8
 versions
 interoperability 34-8
 troubleshooting interoperability problems 34-21
 v2 improvements 34-3
ping
 executing 36-10
 overview 36-10
police aggregate command 30-40
police command
 guidelines 30-39
 with individual policers 30-36
policer aggregate command 30-39
policers
 configuring for more than one traffic class 30-39
 described 30-2
policing
 aggregate in input policy maps 30-14
 described 30-2
 individual in input policy maps 30-13
 priority in output policy maps 30-15
 QoS 30-12
policy-based routing
 See PBR
policy-map command 30-3
policy-map marking, configuration guidelines 30-41
policy maps
 attaching 30-4, 30-35
 configuration examples 30-56
 described 30-13
 input
 configuring 30-35
 described 30-4
 output
 configuring 30-43
 described 30-4
port ACLs
 defined 28-2
 types of 28-3
Port Aggregation Protocol
 See EtherChannel
port-based authentication
 accounting 8-5
 authentication server
 defined 8-2
 RADIUS server 8-2
 client, defined 8-2
 configuration guidelines 8-10
 configuring
 802.1x authentication 8-11
 host mode 8-17
 manual re-authentication of a client 8-14
 periodic re-authentication 8-13
 quiet period 8-14
 RADIUS server 8-13

port-based authentication, configuring (continued)
RADIUS server parameters on the switch 8-12
switch-to-client frame-retransmission number 8-15, 8-16
switch-to-client retransmission time 8-15
default configuration 8-9
described 8-1
device roles 8-2
displaying statistics 8-19
EAPOL-start frame 8-3
EAP-request/identity frame 8-3
EAP-response/identity frame 8-3
encapsulation 8-3
host mode 8-6
initiation and message exchange 8-3
method lists 8-11
multiple-hosts mode, described 8-6
ports
authorization state and dot1x port-control command 8-4
authorized and unauthorized 8-4
port security
described 8-7
interactions 8-7
multiple-hosts mode 8-6
resetting to default values 8-17
statistics, displaying 8-19
switch
as proxy 8-3
RADIUS client 8-3
VLAN assignment
AAA authorization 8-11
characteristics 8-8
configuration tasks 8-8
described 8-8
port blocking 1-2, 21-7
port-channel
See EtherChannel
Port Fast
described 16-2
enabling 16-5
support for 1-4
port membership modes, VLAN 11-4
port priority
MSTP 15-17
STP 14-16
ports
access 9-3
blocking 21-7
dynamic access 11-5
IEEE 802.1Q tunnel 11-5
protected 21-5
routed 9-4
secure 21-8
static-access 11-4, 11-10
switch 9-2
trunks 11-4, 11-14
VLAN assignments 11-10
port security
aging 21-15
configuring 21-11
default configuration 21-10
described 21-8
displaying 21-17
on trunk ports 21-12
sticky learning 21-9
violations 21-9
with other features 21-10
port shaping
configuring 30-47
described 30-20
port-shutdown response, VMPS 11-23
preferential treatment of traffic
See QoS
prefix lists, BGP 32-50
preventing unauthorized access 7-1
primary links 17-1

primary VLANs 12-2, 12-3
priority
 HSRP 33-6
priority command 30-15
 configuring strict priority queuing 30-48
 for QoS scheduling 30-18
 for strict priority queuing 30-22
priority policing, described 30-15
priority queues
 configuring 30-48
 described 30-22
 for QoS scheduling 30-18
priority with police
 commands 30-16
 configuring 30-50
 described 30-22
priority with unconditional policing, QoS 30-18
private VLANs
 across multiple switches 12-4
 and SVIs 12-5
 benefits of 12-1
 community ports 12-3
 community VLANs 12-2, 12-3
 configuration guidelines 12-6, 12-7, 12-8
 configuration tasks 12-6
 configuring 12-10
 default configuration 12-6
 end station access to 12-3
 IP addressing 12-4
 isolated port 12-3
 isolated VLANs 12-2, 12-3
 mapping 12-14
 monitoring 12-15
private VLANs (continued)
 ports
 community 12-3
 configuration guidelines 12-8
 configuring host ports 12-12
 configuring promiscuous ports 12-13
 described 11-5
 isolated 12-3
 promiscuous 12-2
 primary VLANs 12-2, 12-3
 promiscuous ports 12-2
 secondary VLANs 12-2
 subdomains 12-1
 traffic in 12-5
privileged EXEC mode 2-2
privilege levels
 changing the default for lines 7-9
 exiting 7-10
 logging into 7-10
 overview 7-2, 7-8
 setting a command with 7-8
promiscuous ports
 configuring 12-13
 defined 12-2
protected ports 21-5
protocol-dependent modules, EIGRP 32-33
Protocol-Independent Multicast Protocol
 See PIM
provider edge devices 1-14, 32-59
proxy ARP
 configuring 32-10
 definition 32-8
 with IP routing disabled 32-10
PVST+
 802.1Q trunking interoperability 14-10
 described 14-9
 instances supported 14-10

Q

QoS
- aggregate policers 30-14
- and MQC 30-1
- basic model 30-2
- CBWFQ 30-21
- CBWFQ, configuring 30-44
- class-based shaping, described 30-19
- classification
 - ACL lookup 30-10
 - based on CoS value 30-7
 - based on DSCP 30-8
 - based on IP precedence 30-8
 - based on QoS group 30-10
 - class maps, described 30-6
 - comparisons 30-9
 - criteria 30-5
 - in frames and packets 30-6
 - policy maps, described 30-13
- class maps, configuration guidelines 30-31
- class maps, configuring 30-31
- configuration examples
 - adding customers 30-58
 - adding or deleting a class 30-61
 - adding or deleting classification criteria 30-58, 30-59
 - adding or deleting configured actions 30-60
 - changing queuing or scheduling parameters 30-60
- configuration guidelines
 - aggregate policers 30-39
 - CBWFQ 30-44
 - class-based shaping 30-46
 - class maps 30-31
 - general 30-27
 - individual policers 30-36
 - input policy maps 30-35
 - marking 30-41
 - output policy maps 30-43

QoS, configuration guidelines (continued)
 - unconditional priority policing 30-50
 - WTD 30-53
- configuring
 - aggregate policers 30-39
 - class-based shaping 30-46
 - classification with IP ACLs 30-27
 - class maps 30-31, 30-32
 - individual policers 30-36
 - individual policing 30-36
 - input policy maps with marking 30-41
 - IP ACLs 30-27
 - MAC ACLs 30-30
 - output policy maps 30-43
 - port shaping 30-47
 - priority queues 30-48
 - queue size 30-24
 - requirements 30-26
 - service policies 30-35
 - strict priority queuing 30-48
 - table maps 30-34
 - unconditional priority policing 30-50
 - WTD 30-53
- congestion avoidance 30-2, 30-24
- congestion management 30-2, 30-18
- default configuration 30-27
- initial configuration example 30-56
- input policy maps
 - configuring 30-35
 - described 30-4
- IP packet classification 30-5
- Layer 2 packet classification 30-5
- Layer 3 packet classification 30-5
- marking, described 30-2
- match command 30-7
- output policy maps
 - configuring 30-44
 - described 30-5
- overview 30-1

QoS (continued)
packet classification 30-2
packet marking 30-16
packet policing 30-2
parent-child hierarchy 30-20
policers
configuring 30-37, 30-40, 30-51
described 30-12
policing
aggregate 30-14
described 30-2, 30-12
individual 30-13
priority 30-15
policy maps
attaching 30-35
attaching to an interface 30-15
displaying statistics 30-55
port shaping, described 30-20
preconfiguration 30-26
priority policing, described 30-15
priority with police 30-22
queue size 30-24
scheduling 30-18
CBWFQ 30-18
priority queuing 30-18
traffic shaping 30-18
strict priority queuing 30-22
supported table maps 30-12
support for 1-6
table maps 30-11
traffic shaping, described 30-19
unconditional priority policing 30-22
WTD 30-24
QoS groups
classification 30-10
described 30-5, 30-10
QoS information, displaying 30-55
quality of service
See QoS
queries, IGMP 20-3
query solicitation, IGMP 20-11
queue bandwidth and queue size, relationship 30-25
queue-limit command, QoS 30-24, 30-25, 30-53
queue size, QoS, managing 30-24

R

RADIUS
attributes
vendor-proprietary 7-31
vendor-specific 7-29
configuring
accounting 7-28
authentication 7-23
authorization 7-27
communication, global 7-21, 7-29
communication, per-server 7-20, 7-21
multiple UDP ports 7-21
default configuration 7-20
defining AAA server groups 7-25
displaying the configuration 7-31
identifying the server 7-20
limiting the services to the user 7-27
method list, defined 7-20
operation of 7-19
overview 7-18
suggested network environments 7-18
support for 1-6
tracking services accessed by user 7-28
range
macro 9-10
of interfaces 9-8
rapid convergence 15-7
rapid per-VLAN spanning-tree plus
See rapid PVST+

rapid PVST+
802.1Q trunking interoperability 14-10
described 14-9
instances supported 14-10
Rapid Spanning Tree Protocol
See RSTP
RARP 32-8
RCP
configuration files
downloading B-16
overview B-15
preparing the server B-15
uploading B-17
image files
deleting old image B-31
downloading B-29
preparing the server B-28
uploading B-31
reconfirmation interval, VMPS, changing 11-27
reconfirming dynamic VLAN membership 11-26
recovery procedures 36-1
redundancy
EtherChannel 31-3
HSRP 33-1
STP
backbone 14-8
path cost 11-21
port priority 11-20
reliable transport protocol, EIGRP 32-33
reloading software 3-16
Remote Authentication Dial-In User Service
See RADIUS
Remote Copy Protocol
See RCP
Remote Network Monitoring
See RMON
Remote SPAN
See RSPAN
remote SPAN 24-2
report suppression, IGMP
described 20-6
disabling 20-14
resequencing ACL entries 28-14
resets, in BGP 32-45
resetting a UDLD-shutdown interface 23-6
restricting access
NTP services 5-8
overview 7-1
passwords and privilege levels 7-2
RADIUS 7-18
TACACS+ 7-10
retry count, VMPS, changing 11-27
reverse address resolution 32-7
Reverse Address Resolution Protocol
See RARP
RFC
1058, RIP 32-18
1112, IP multicast and IGMP 20-2
1157, SNMPv1 27-2
1163, BGP 32-38
1166, IP addresses 32-5
1253, OSPF 32-23
1267, BGP 32-38
1305, NTP 5-2
1587, NSSAs 32-24
1757, RMON 25-2
1771, BGP 32-38
1901, SNMPv2C 27-2
1902 to 1907, SNMPv2 27-2
2236, IP multicast and IGMP 20-2
2273-2275, SNMPv3 27-2
2475, DSCP 30-9
2597, AF per-hop behavior 30-9
2598, EF 30-9

RIP
advertisements 32-18
authentication 32-21
configuring 32-19
default configuration 32-19
described 32-18
hop counts 32-18
split horizon 32-22
summary addresses 32-22
support for 1-7
RMON
default configuration 25-3
displaying status 25-6
enabling alarms and events 25-3
groups supported 25-2
overview 25-1
statistics
collecting group Ethernet 25-6
collecting group history 25-5
support for 1-8
root guard
described 16-3
enabling 16-9
support for 1-4
root switch
MSTP 15-14
STP 14-14
route calculation timers, OSPF 32-29
route dampening, BGP 32-57
routed packets, ACLs on 28-38
routed ports
configuring 32-3
defined 9-4
IP addresses on 9-20, 32-3
route-map command 32-78
route maps
BGP 32-48
policy-based routing 32-77
router ACLs
defined 28-2
types of 28-4
route reflectors, BGP 32-56
router ID, OSPF 32-31
route selection, BGP 32-46
route summarization, OSPF 32-29
route targets, VPN 32-61
routing
default 32-2
dynamic 32-2
redistribution of information 32-73
static 32-2
routing domain confederation, BGP 32-55
Routing Information Protocol
See RIP
routing protocol administrative distances 32-71
RSPAN 24-2
characteristics 24-8
configuration guidelines 24-16
default configuration 24-10
destination ports 24-7
displaying status 24-22
interaction with other features 24-8
monitored ports 24-5
monitoring ports 24-7
overview 1-8, 24-1
received traffic 24-4
session limits 24-10
sessions
creating 24-17
defined 24-3
limiting source traffic to specific VLANs 24-21
specifying monitored ports 24-17
with ingress traffic enabled 24-20
source ports 24-5
transmitted traffic 24-5
VLAN-based 24-6

RSTP
active topology 15-6
BPDU
format 15-9
processing 15-10
designated port, defined 15-6
designated switch, defined 15-6
interoperability with 802.1D
described 15-5
restarting migration process 15-22
topology changes 15-10
overview 15-6
port roles
described 15-6
synchronized 15-8
proposal-agreement handshake process 15-7
rapid convergence
described 15-7
edge ports and Port Fast 15-7
point-to-point links 15-7, 15-22
root ports 15-7
root port, defined 15-6
See also MSTP
running configuration, saving 3-10

S

scheduled reloads 3-16
scheduling, QoS 30-18
SDM
described 6-1
templates
configuring 6-3
number of 6-1
SDM template
configuration guidelines 6-2
configuring 6-2
default 6-1
layer 2 6-1
types of 6-1
secondary VLANs 12-2
secure MAC addresses
deleting 21-14
maximum number of 21-9
types of 21-8
secure ports, configuring 21-8
secure remote connections 7-38
Secure Shell
See SSH
security, port 21-8
security features 1-5
sequence numbers in log messages 26-7
service-policy command
attaching policy maps 30-4
guidelines 30-43
using 30-35
service-provider network, MSTP and RSTP 15-1
service-provider networks
and customer VLANs 13-2
and IEEE 802.1Q tunneling 13-1
Layer 2 protocols across 13-8
Layer 2 protocol tunneling for EtherChannels 13-9
set command
for QoS marking 30-16
guidelines 30-41
set-request operation 27-4
severity levels, defining in system messages 26-8
SFPs
monitoring status of 1-8, 9-23, 36-9
security and identification 36-9
status, displaying 1-8

shape average command, QoS 30-18, 30-20, 30-46
shaped round robin
See SRR
show access-lists hw-summary command 28-21
show and more command output, filtering 2-8
show cdp traffic command 22-5
show configuration command 9-19
show forward command 36-19
show interfaces command 9-16, 9-19
show l2protocol command 13-13, 13-15, 13-16
show platform forward command 36-19
show running-config command
displaying ACLs 28-19, 28-20, 28-31, 28-33
interface description in 9-19
shutdown command on interfaces 9-24
shutdown threshold for Layer 2 protocol packets 13-11
Simple Network Management Protocol
See SNMP
Smartports macros
applying global parameter values 10-4
applying macros 10-4
applying parameter values 10-4
configuration guidelines 10-2
creating 10-3
default configuration 10-2
defined 10-1
displaying 10-5
tracing 10-2
website 10-1
SNAP 22-1
SNMP
accessing MIB variables with 27-4
agent
described 27-4
disabling 27-7
authentication level 27-10
community strings
configuring 27-8
overview 27-4
SNMP (continued)
configuration examples 27-15
default configuration 27-6
engine ID 27-7
groups 27-6, 27-9
host 27-6
ifIndex values 27-5
in-band management 1-4
informs
and trap keyword 27-11
described 27-5
differences from traps 27-5
disabling 27-14
enabling 27-14
limiting access by TFTP servers 27-15
limiting system log messages to NMS 26-9
manager functions 1-3, 27-3
MIBs
location of A-3
supported A-1
notifications 27-5
overview 27-1, 27-4
security levels 27-3
status, displaying 27-16
system contact and location 27-14
trap manager, configuring 27-13
traps
described 27-3, 27-5
differences from informs 27-5
disabling 27-14
enabling 27-11
enabling MAC address notification 5-22
overview 27-1, 27-4
types of 27-11
users 27-6, 27-9
versions supported 27-2
SNMPv1 27-2
SNMPv2C 27-2
SNMPv3 27-2

snooping, IGMP 20-1
software images
location in flash B-19
recovery procedures 36-2
scheduling reloads 3-17
tar file format, described B-19
See also downloading and uploading
source addresses, in IPv4 ACLs 28-11
source-and-destination-IP address based forwarding, EtherChannel 31-8
source-and-destination MAC address forwarding, EtherChannel 31-7
source-IP address based forwarding, EtherChannel 31-7
source-MAC address forwarding, EtherChannel 31-7
SPAN
configuration guidelines 24-10
default configuration 24-10
destination ports 24-7
displaying status 24-22
interaction with other features 24-8
monitored ports 24-5
monitoring ports 24-7
overview 1-8, 24-1
ports, restrictions 21-11
received traffic 24-4
session limits 24-10
sessions
configuring ingress forwarding 24-14, 24-21
creating 24-11
defined 24-3
limiting source traffic to specific VLANs 24-15
removing destination (monitoring) ports 24-12
specifying monitored ports 24-11
with ingress traffic enabled 24-13
source ports 24-5
transmitted traffic 24-5
VLAN-based 24-6
spanning tree and native VLANs 11-15
Spanning Tree Protocol
See STP
SPAN traffic 24-4
speed, configuring on interfaces 9-15
split horizon, RIP 32-22
SRR, support for 1-7
SSH
configuring 7-39
cryptographic software image 7-37
described 1-4, 7-38
encryption methods 7-38
user authentication methods, supported 7-38
standby ip command 33-5
standby links 17-1
standby router 33-1
standby timers, HSRP 33-9
startup configuration
booting
manually 3-14
specific image 3-14
clearing B-18
configuration file
automatically downloading 3-13
specifying the filename 3-13
default boot configuration 3-13
static access ports
assigning to VLAN 11-10
defined 9-3, 11-4
static addresses
See addresses
static IP routing 1-7
static MAC addressing 1-6
static routes
configuring 32-71
static routing 32-2
static VLAN membership 11-2

statistics
IEEE 802.1x 8-19
CDP 22-5
interface 9-23
IP multicast routing 34-34
OSPF 32-31
RMON group Ethernet 25-6
RMON group history 25-5
SNMP input and output 27-16
sticky learning 21-9
storm control
configuring 21-3
described 21-1
disabling 21-5
displaying 21-17
support for 1-2
thresholds 21-1
STP
BPDU filtering
described 16-3
disabling 16-8
enabling 16-7
BPDU guard
described 16-2
disabling 16-7
enabling 16-6
BPDU message exchange 14-3
configuration guidelines 14-12, 16-5
configuring
forward-delay time 14-21
hello time 14-20
maximum aging time 14-21
path cost 14-18
port priority 14-16
root switch 14-14
secondary root switch 14-16
spanning-tree mode 14-13
switch priority 14-19
counters, clearing 14-22
STP (continued)
default configuration 14-11
default optional feature configuration 16-5
designated port, defined 14-4
designated switch, defined 14-4
disabling 14-14
displaying status 14-22
EtherChannel guard
described 16-3
disabling 16-9
enabling 16-8
extended system ID
effects on root switch 14-14
effects on the secondary root switch 14-16
overview 14-4
unexpected behavior 14-15
features supported 1-4
IEEE 802.1D and bridge ID 14-4
IEEE 802.1D and multicast addresses 14-8
IEEE 802.1t and VLAN identifier 14-4
inferior BPDU 14-3
instances supported 14-10
interface state, blocking to forwarding 16-2
interface states
blocking 14-6
disabled 14-7
forwarding 14-6, 14-7
learning 14-6
listening 14-6
overview 14-4
interoperability and compatibility among modes 14-10
keepalive messages 14-3
Layer 2 protocol tunneling 13-8
limitations with 802.1Q trunks 14-10
load sharing
overview 11-19
using path costs 11-21
using port priorities 11-20

STP (continued)
loop guard
described 16-4
enabling 16-9
modes supported 14-9
multicast addresses, effect of 14-8
optional features supported 1-4
overview 14-2
path costs 11-21, 11-22
Port Fast
described 16-2
enabling 16-5
port priorities 11-20
preventing root switch selection 16-3
protocols supported 14-9
redundant connectivity 14-8
root guard
described 16-3
enabling 16-9
root port, defined 14-3
root switch
configuring 14-14
effects of extended system ID 14-4, 14-14
election 14-3
unexpected behavior 14-15
shutdown Port Fast-enabled port 16-2
status, displaying 14-22
superior BPDU 14-3
timers, described 14-20
stratum, NTP 5-2
strict priority queuing 30-48
configuration guidelines 30-48
configuring 30-49
defined 30-22
QoS 30-22
stub areas, OSPF 32-27
subdomains, private VLAN 12-1
subnet mask 32-5
subnet zero 32-6
success response, VMPS 11-23
summer time 5-13
SunNet Manager 1-3
supernet 32-6
SVIs
and IP unicast routing 32-3
and router ACLs 28-4
connecting VLANs 9-6
defined 9-5
routing between VLANs 11-2
switch console port 1-4
Switch Database Management
See SDM
switched packets, ACLs on 28-37
Switched Port Analyzer
See SPAN
switched ports 9-2
switchport block multicast command 21-7
switchport block unicast command 21-7
switchport command 9-12
switchport mode dot1q-tunnel command 13-7
switchport protected command 21-6
switch priority
MSTP 15-19
STP 14-19
switch software features 1-1
switch virtual interface
See SVI
synchronization, BGP 32-43
syslog
See system message logging
system clock
configuring
daylight saving time 5-13
manually 5-11
summer time 5-13
time zones 5-12
displaying the time and date 5-12
overview 5-2

system clock
See also NTP
system message logging
default configuration 26-3
defining error message severity levels 26-8
disabling 26-3
displaying the configuration 26-12
enabling 26-4
facility keywords, described 26-11
level keywords, described 26-9
limiting messages 26-9
message format 26-2
overview 26-1
sequence numbers, enabling and disabling 26-7
setting the display destination device 26-4
synchronizing log messages 26-5
syslog facility 1-8
time stamps, enabling and disabling 26-7
UNIX syslog servers
configuring the daemon 26-10
configuring the logging facility 26-11
facilities supported 26-11
system MTU and IEEE 802.1Q tunneling 13-5
system name
default configuration 5-15
default setting 5-15
manual configuration 5-15
See also DNS
system prompt
default setting 5-14 to 5-15
system resources, optimizing 6-1

T

table maps
default actions 30-11
described 30-11
for QoS marking 30-16
QoS, configuring 30-34
types of 30-12
TACACS+
accounting, defined 7-11
authentication, defined 7-11
authorization, defined 7-11
configuring
accounting 7-17
authentication key 7-13
authorization 7-16
login authentication 7-14
default configuration 7-13
displaying the configuration 7-17
identifying the server 7-13
limiting the services to the user 7-16
operation of 7-12
overview 7-10
support for 1-6
tracking services accessed by user 7-17
tagged packets
IEEE 802.1Q 13-3
Layer 2 protocol 13-8
tar files
creating B-6
displaying the contents of B-6
extracting B-7
image file format B-19
TDR 1-8
Telnet
accessing management interfaces 2-9
number of connections 1-4
setting a password 7-6

templates, SDM 6-2
Terminal Access Controller Access Control System Plus
 See TACACS+
terminal lines, setting a password 7-6
TFTP
 configuration files
 downloading B-10
 preparing the server B-10
 uploading B-11
 configuration files in base directory 3-6
 configuring for autoconfiguration 3-5
 image files
 deleting B-22
 downloading B-21
 preparing the server B-21
 uploading B-23
 limiting access by servers 27-15
TFTP server 1-3
threshold, traffic level 21-2
time
 See NTP and system clock
Time Domain Reflector
 See TDR
time-range command 28-16
time ranges in ACLs 28-16
time stamps in log messages 26-7
time zones 5-12
traceroute, Layer 2
 and ARP 36-14
 and CDP 36-14
 broadcast traffic 36-13
 described 36-13
 IP addresses and subnets 36-14
 MAC addresses and VLANs 36-14
 multicast traffic 36-14
 multiple devices on a port 36-14
 unicast traffic 36-13
 usage guidelines 36-14
traceroute command 36-16
 See also IP traceroute
traffic
 blocking flooded 21-7
 fragmented 28-5
 unfragmented 28-5
traffic class, defined 30-3
traffic classification, typical values 30-9
traffic marking 30-16
traffic policies, elements in 30-3
traffic shaping
 for QoS scheduling 30-18
 QoS traffic control 30-19
traffic suppression 21-1
trap-door mechanism 3-2
traps
 configuring MAC address notification 5-22
 configuring managers 27-11
 defined 27-3
 enabling 5-22, 27-11
 notification types 27-11
 overview 27-1, 27-4
troubleshooting
 connectivity problems 36-10, 36-13, 36-15
 detecting unidirectional links 23-1
 displaying crash information 36-22
 PIMv1 and PIMv2 interoperability problems 34-21
 setting packet forwarding 36-19
 SFP security and identification 36-9
 show forward command 36-19
 with CiscoWorks 27-4
 with debug commands 36-18
 with ping 36-10
 with system message logging 26-1
 with traceroute 36-15
trunking encapsulation 1-5
trunk ports
 configuring 11-17
 defined 9-4, 11-4

trunks
allowed-VLAN list 11-17
load sharing
setting STP path costs 11-21
using STP port priorities 11-20
native VLAN for untagged traffic 11-19
parallel 11-21
tunneling
defined 13-1
IEEE 802.1Q 13-1
Layer 2 protocol 13-8
tunnel ports
defined 11-5
described 9-4, 13-1
IEEE 802.1Q, configuring 13-6
incompatibilities with other features 13-6
twisted-pair Ethernet, detecting unidirectional links 23-1

U

UDLD
configuration guidelines 23-4
default configuration 23-4
disabling
globally 23-5
on fiber-optic interfaces 23-5
per interface 23-5
echoing detection mechanism 23-3
enabling
globally 23-5
per interface 23-5
Layer 2 protocol tunneling 13-10
link-detection mechanism 23-1
neighbor database 23-2
overview 23-1
resetting an interface 23-6
status, displaying 23-6
support for 1-4
UDP, configuring 32-14
unauthorized ports with 802.1x 8-4
unconditional priority policing
configuration guidelines 30-50
priority with police 30-22
UNI
configuring 9-13
described 9-3
unicast MAC address filtering 1-3
and adding static addresses 5-26
and broadcast MAC addresses 5-25
and CPU packets 5-25
and multicast addresses 5-25
and router MAC addresses 5-25
configuration guidelines 5-25
described 5-25
unicast storm 21-1
unicast storm control command 21-4
unicast traffic, blocking 21-7
UniDirectional Link Detection protocol
See UDLD
UNIX syslog servers
daemon configuration 26-10
facilities supported 26-11
message logging configuration 26-11
upgrading information
See release notes
upgrading software images
See downloading
uploading
configuration files
preparing B-10, B-12, B-15
reasons for B-8
using FTP B-14
using RCP B-17
using TFTP B-11

uploading (continued)
image files
preparing B-21, B-24, B-28
reasons for B-19
using FTP B-27
using RCP B-31
using TFTP B-23
User Datagram Protocol
See UDP
user EXEC mode 2-2
username-based authentication 7-7
user network interface
See UNI

V

Virtual Private Network
See VPN
virtual router 33-1, 33-2
vlan.dat file 11-3
VLAN 1
disabling on a trunk port 11-18
minimization 11-18
VLAN ACLs
See VLAN maps
vlan-assignment response, VMPS 11-23
VLAN configuration mode 2-2
VLAN database
VLAN configuration saved in 11-9
VLANs saved in 11-3
vlan dot1q tag native command 13-5
VLAN filtering and SPAN 24-6
vlan global configuration command 11-6, 11-9
VLAN ID, discovering 5-28
VLAN Management Policy Server
See VMPS
VLAN map entries, order of 28-30
VLAN maps
applying 28-33
common uses for 28-34
configuration guidelines 28-30
configuring 28-29
creating 28-31
defined 28-2, 28-5
denying access to a server example 28-35
denying and permitting packets 28-31
displaying 28-40
examples of ACLs and VLAN maps 28-31
removing 28-33
support for 1-6
wiring closet configuration example 28-34
with router ACLs 28-39
VLAN membership
confirming 11-26
modes 11-4
VLAN Query Protocol
See VQP
VLANs
adding 11-9
aging dynamic addresses 14-9
allowed on trunk 11-17
and spanning-tree instances 11-3, 11-8
configuration guidelines 11-8
configuration guidelines, normal-range VLANs 11-8
configuring 11-1
connecting through SVIs 9-6
customer numbering in service-provider networks 13-3
default configuration 11-7
described 9-2, 11-1
displaying 11-14
extended-range 11-1
features 1-5
illustrated 11-2
internal 11-8
limiting source traffic with RSPAN 24-21
limiting source traffic with SPAN 24-15

VLANs (continued)
modifying 11-9
multicast 20-16
native, configuring 11-19
normal-range 11-1, 11-3
number supported 1-5
parameters 11-3
port membership modes 11-4
static-access ports 11-10
STP and 802.1Q trunks 14-10
supported 11-3
traffic between 11-2
VLAN trunks 11-14
VMPS
administering 11-27
configuration example 11-28
configuration guidelines 11-25
default configuration 11-24
description 11-23
dynamic port membership
described 11-24
reconfirming 11-27
troubleshooting 11-28
mapping MAC addresses to VLANs 11-23
monitoring 11-27
reconfirmation interval, changing 11-27
reconfirming membership 11-26
retry count, changing 11-27
VPN
configuring routing in 32-63
forwarding 32-61
in service provider networks 32-59
routes 1-14, 32-59
VPN routing and forwarding table
See VRF
VQP 1-5, 11-23
VRF
defining 32-61
tables 1-14, 32-59
VTP
Layer 2 protocol tunneling 13-8

W

weighted tail drop
See WTD
WTD
configuration guidelines 30-53
configuring 30-53
described 30-24
support for 1-7

X

Xmodem protocol 36-2